THE LISTER HILL NATIONAL CENTER
FOR BIOMEDICAL COMMUNICATIONS

A research division of the U.S. National Library of Medicine

LHNCBC-TR-2008-001

Automation to Accelerate the Production of MEDLINE

April 2008

George R. Thoma, Ph.D.
Daniel Le, Ph.D.

Incheol Kim, Ph.D, Jong Woo Kim, Ph.D.
Chan Moon, Loc Tran, Jie Zou, Ph.D.

U.S. National Library of Medicine, LHNCBC
8600 Rockville Pike, Building 38A
Bethesda, MD 20894

TABLE OF CONTENTS

AUTOMATION TO ACCELERATE THE PRODUCTION OF MEDLINE

1. Background

Containing 16 million citations to the biomedical journal literature, MEDLINE® is accessed more than 3 million times a day worldwide. The rapid increase in both the number of journals indexed and the number of citations produced are evident from Figures 1.1 and 1.2. The number of journals indexed by NLM has increased by about 130 titles a year (on the average) over the past ten years, and now is approximately 5,200. The increase in the number of citations is even more dramatic: extrapolating the curve in Figure 1.2 suggests that at the current rate, the number of citations produced annually will amount to almost 700,000 in 2008, and exceed a million in a few short years, creating possible pressures on NLM's indexing budget. This has motivated research and development toward finding ways to automate the process of acquiring bibliographic data, as well as the indexing process itself. Bibliographic data describing the articles to be indexed consist of more than 50 fields (e.g., author names, article title, affiliation, abstract, journal name, page numbers, etc.), and these are assembled and presented to indexers who add keywords and Medical Subject Heading (MeSH) terms.

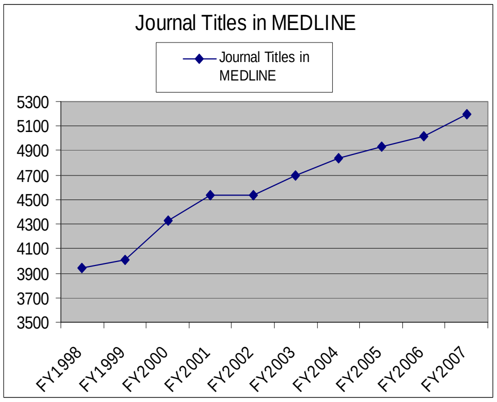


Figure 1.1

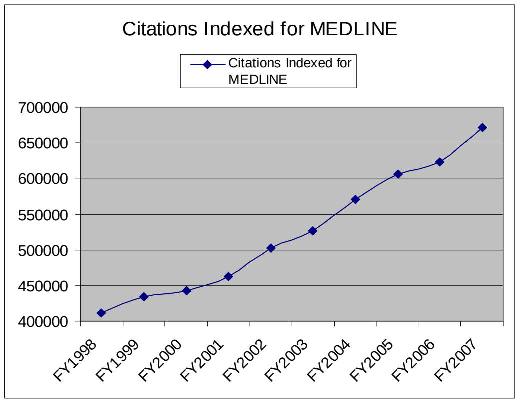


Figure 1.2

There are two ways in which bibliographic data is provided to the indexers. The first is from a system (MARS for *Medical Article Records System*) built and maintained by the Lister Hill Center's Communications Engineering Branch that involves scanning and processing paper journals for the automated extraction of four key fields that represent over 90% of the text in the citation. The main components of MARS are scanning and OCR, followed by rule-based algorithms for page segmentation (zoning), labeling the zones, pattern matching to extract the citation data, and reformatting zone contents to adhere to MEDLINE conventions. The introduction of MARS allowed NLM to discontinue the manual entry of the bibliographic data, the traditional (and expensive) practice followed for decades.

The second way bibliographic data is currently obtained is directly from publishers who send these to NLM in XML format. While publishers include much of the required data in their submissions, they omit several data-intensive fields (e.g., databank accession numbers, grant numbers) presumably because the inclusion of these fields would be highly labor-intensive: the publisher staff would have to manually read through every article, and find and record these fields. Since manual entry of this data would be equally burdensome for NLM staff, we are meeting this challenge by developing the *Publisher Data Review* (PDR) system to extract these fields directly from the online articles on publishers' Web sites, as well as to correct errors in the XML files submitted, thereby considerably reducing the manual effort and accelerating the production of MEDLINE citations.

While the focus of this report is our development of supervised machine learning algorithms to automatically extract these fields, we may note that the challenge in incorporating these algorithms in a production environment also includes the development of an effective work distribution system and intuitive user interfaces.

This report is organized as follows. Following a brief statement of project objectives and significance, we present in Section 4 an overall description of the PDR system, in addition to defining the four items of bibliographic data extracted by PDR. In Section 5, the input and output subsystems of PDR are described. In Sections 6 to 8, we describe the algorithmic research and design that underlies the automated functions of PDR. Finally, Section 9 summarizes the evaluation of our algorithms and Section 10 outlines ongoing and future steps.

2. Project Objectives

The overall goal of the project is to help accelerate the production of MEDLINE citations and thereby control their cost. Our specific objective is to employ advanced machine learning techniques to automatically extract bibliographic data from medical articles.

3. Project Significance

The placing of MARS in production to automatically extract bibliographic data from scanned journals has allowed NLM to realize significant savings by eliminating contracts for the manual entry of citation data. Similarly, the implementation of PDR to automatically extract key bibliographic data from online journals is expected to result in further savings. Moreover, the design of algorithms for the automated extraction of bibliographic data can inform similar development of modules in other ongoing projects, e.g., digital preservation of historic documents, in which descriptive metadata is key to accessing and using these documents preserved for the long term. The automated extraction of such metadata is essential to making digital preservation affordable.

4. Publisher Data Review (PDR) system

4.1 Purpose for PDR

The PDR system provides operators data missing from the XML citations sent in directly by publishers (such as databank accession numbers, grant numbers, granting agencies, and PubMed IDs of articles commented on by authors) thereby reducing the burden on operators in creating citations for MEDLINE. Correcting the publisher data is currently a labor-intensive process since the operators perform these functions manually by looking through an entire article to find these items, and then keying them in.

4.2 PDR System description and overview

The system consists of five automated subsystems and a client-based "reconcile" (text verification) subsystem as shown in Figure 4.1. All subsystems are networked via a LAN and communicate through a PDR database server and an XML file server.

Briefly, the PDR system works as follows. The **Get Citations from DCMS Queue** subsystem periodically retrieves publisher-supplied XML citation files from DCMS (Data Creation and Maintenance System), a database maintained by NLM's OCCS division[1]. DCMS communicates with publisher Web sites over the Internet to receive and store these publisher-supplied XML files. The **HTML/PDF Files Download** subsystem processes these XML files to obtain article links which are used to connect to the publisher Web sites and download full text article files. This subsystem also converts articles in PDF format to HTML, and validates them as full text articles, rather than abstracts or summaries (Section 5).

The **Zone Creation** subsystem segments the articles to create zones based on geometric layout, the recursive X-Y cut algorithm, and HTML Document Object Model (Section 6). The **Zone Labeling** subsystem labels and ranks these zones with appropriate field labels such as *databank accession numbers, grant numbers*, and *grant support*, using Naïve Bayesian and Support Vector Machine (SVM) algorithms (Section 7). The **Bibliographic Data Extraction** subsystem discards irrelevant contents in the labeled zones, and extracts bibliographic items using a hybrid contextual and statistical method and the SVM algorithm (Section 8). Finally, the **Client-based PDR Reconcile** subsystem, which is activated by operators through another NLM subsystem named **Client-based DCMS**, presents the extracted bibliographic data to operators for verification before they are uploaded to DCMS and made available to indexers (Section 5).

[1] DCMS may be viewed as an intermediate database for bibliographic data of articles to be indexed. NLM's indexers view this data, modify items as necessary, and add MeSH and other key terms to complete citations. These citations are then transferred to the MEDLINE database from which they are accessed by users through the PubMed interface.

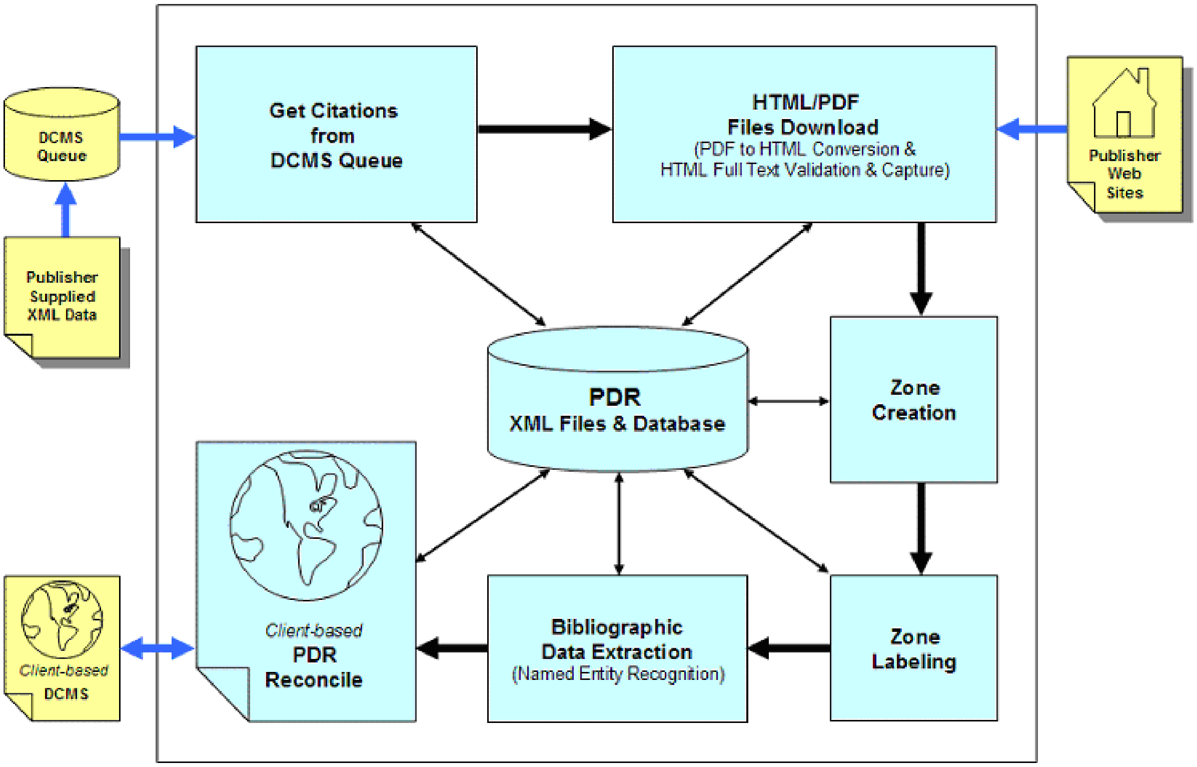

DCMS Queue
Publisher Supplied XML Data
Get Citations from DCMS Queue
HTML/PDF Files Download
(PDF to HTML Conversion & HTML Full Text Validation & Capture)
Publisher Web Sites
PDR XML Files & Database
Zone Creation
Client-based PDR Reconcile
Client-based DCMS
Bibliographic Data Extraction
(Named Entity Recognition)
Zone Labeling

Figure 4.1. The PDR system

4.3 Definitions of PDR-extracted bibliographic data

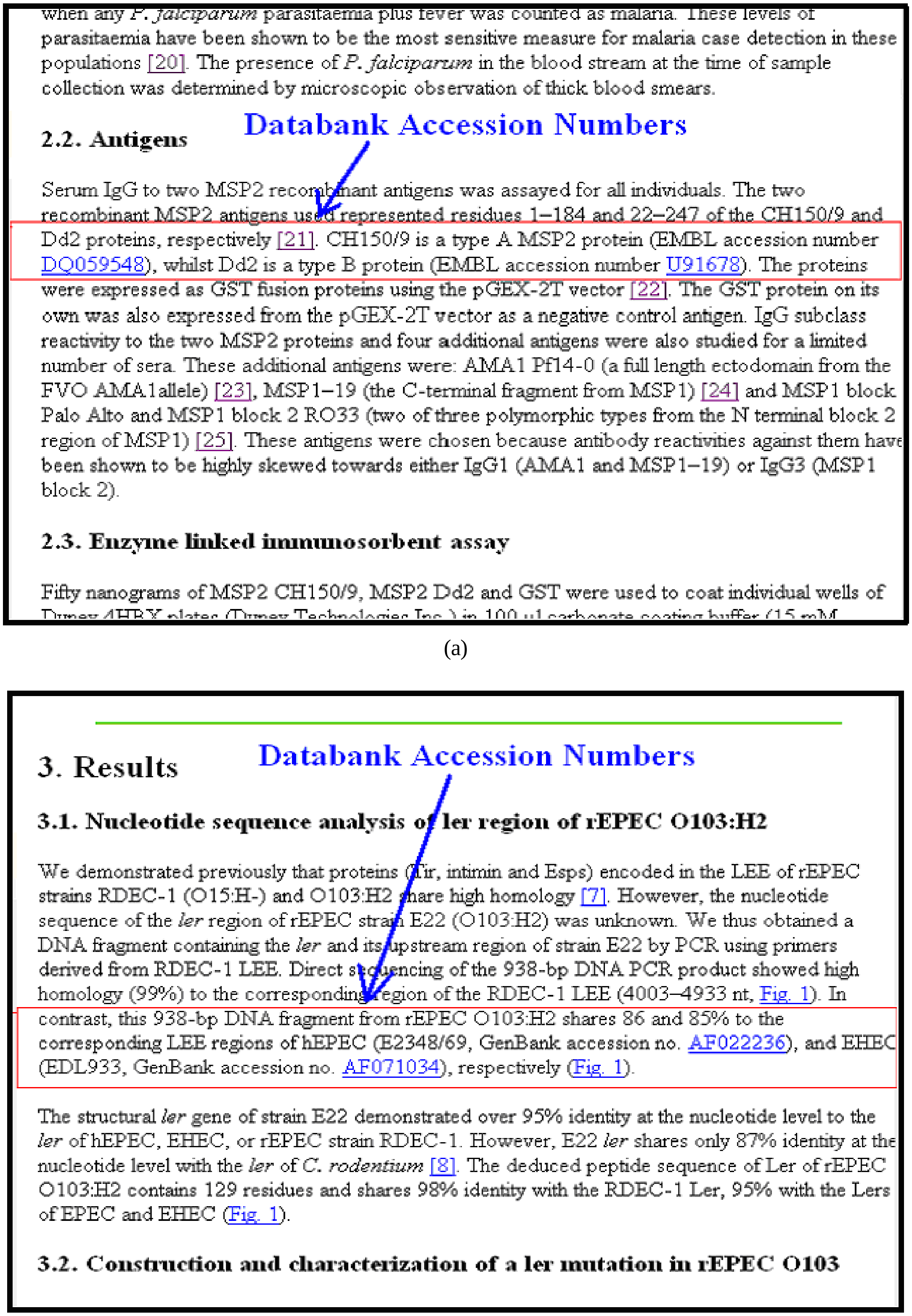
The bibliographic data missing from publisher-supplied XML citations provided by the PDR system include databank accession numbers, grant numbers, grant support (funding institutions), and PubMed IDs (PMIDs) of commented-on articles. The basic definitions of these bibliographic data necessary to understand our algorithms are given here.

Databank Accession Number (DAN)

Databanks register molecular sequence data, gene expression data, clinical trial numbers, or PubChem identifiers. A "databank accession number" (DAN) is the registration number of an entry in one of these databanks. DAN usually appears in an article in close proximity to other information such as the name of a databank and/or words such as "deposit", "submit", etc.

A typical example of a sentence containing DANs is "The PGFS and mPGES-1 sequences reported in this paper have been submitted to the GenBank database under accession numbers AY863054 and AY857634, respectively." In this example, "GenBank" is the databank name and "AY863054' and "AY857634" are DANs.

Figure 4.2 shows examples of articles with DANs. Figure 4.2 (a) shows EMBL as a databank name and DQ059548 and U91678 as DANs. Figure 4.2 (b) shows GenBank as the databank, and AF022236 and AF071034 as DANs.



when any *P. falciparum* parasitaemia plus fever was counted as malaria. These levels of parasitaemia have been shown to be the most sensitive measure for malaria case detection in these populations [20]. The presence of *P. falciparum* in the blood stream at the time of sample collection was determined by microscopic observation of thick blood smears.

2.2. Antigens

Serum IgG to two MSP2 recombinant antigens was assayed for all individuals. The two recombinant MSP2 antigens used represented residues 1–184 and 22–247 of the CH150/9 and Dd2 proteins, respectively [21]. CH150/9 is a type A MSP2 protein (EMBL accession number DQ059548), whilst Dd2 is a type B protein (EMBL accession number U91678). The proteins were expressed as GST fusion proteins using the pGEX-2T vector [22]. The GST protein on its own was also expressed from the pGEX-2T vector as a negative control antigen. IgG subclass reactivity to the two MSP2 proteins and four additional antigens were also studied for a limited number of sera. These additional antigens were: AMA1 Pf14-0 (a full length ectodomain from the FVO AMA1allele) [23], MSP1–19 (the C-terminal fragment from MSP1) [24] and MSP1 block Palo Alto and MSP1 block 2 RO33 (two of three polymorphic types from the N terminal block 2 region of MSP1) [25]. These antigens were chosen because antibody reactivities against them have been shown to be highly skewed towards either IgG1 (AMA1 and MSP1–19) or IgG3 (MSP1 block 2).

2.3. Enzyme linked immunosorbent assay

Fifty nanograms of MSP2 CH150/9, MSP2 Dd2 and GST were used to coat individual wells of Dynex 4HBX plates (Dynex Technologies Inc.) in 100 µl carbonate coating buffer (15 mM

(a)

3. Results

3.1. Nucleotide sequence analysis of ler region of rEPEC O103:H2

We demonstrated previously that proteins (Tir, intimin and Esps) encoded in the LEE of rEPEC strains RDEC-1 (O15:H-) and O103:H2 share high homology [7]. However, the nucleotide sequence of the *ler* region of rEPEC strain E22 (O103:H2) was unknown. We thus obtained a DNA fragment containing the *ler* and its upstream region of strain E22 by PCR using primers derived from RDEC-1 LEE. Direct sequencing of the 938-bp DNA PCR product showed high homology (99%) to the corresponding region of the RDEC-1 LEE (4003–4933 nt, Fig. 1). In contrast, this 938-bp DNA fragment from rEPEC O103:H2 shares 86 and 85% to the corresponding LEE regions of hEPEC (E2348/69, GenBank accession no. AF022236), and EHEC (EDL933, GenBank accession no. AF071034), respectively (Fig. 1).

The structural *ler* gene of strain E22 demonstrated over 95% identity at the nucleotide level to the *ler* of hEPEC, EHEC, or rEPEC strain RDEC-1. However, E22 *ler* shares only 87% identity at the nucleotide level with the *ler* of *C. rodentium* [8]. The deduced peptide sequence of Ler of rEPEC O103:H2 contains 129 residues and shares 98% identity with the RDEC-1 Ler, 95% with the Lers of EPEC and EHEC (Fig. 1).

3.2. Construction and characterization of a ler mutation in rEPEC O103

(b)

Figure 4.2. Examples of Databank Accession Numbers.
In (a) DANs are DQ059548 and U91678. In (b) DANs are AF022236 and AF071034.

Databank	Format	Example
GenBank [1]	[one-letter character][five-digit number], [two-letter character][six-digit number], [three-letter character][five-digit number]	U12345, AF123456 AFC12345
NCT (Clinical Trials) [2]	NCT+[eight-digit number]	NCT 12345678
GEO (Gene Expression Omnibus) [3]	CCC+[any digit number], CCC={GEO, GDS, GSE, GPL, GSM }	GDS01, GSE1234567
ISRCTN [4]	ISRCTN+[eight-digit number],	ISRCTN 12345678

RefSeq
(Reference Sequence) [5]
	CC_[six-digit number], CC_[nine-digit number],
	CC={ AC, AP, NC, NG, NM, NP, NR, NT,
	NW, NZ, XM, XP, XR, YP, ZP }
		AC_123456,
		AC_123456789

OMIM (Online Mendelian Inheritance in Man) [6]	OMIM+$&[five-digit number], $ = { *,#,+,%,^, space}, & = {1,2,3,4,5,6}	OMIM ^123456,
PDB (Protein Data Bank) [7]	[one-digit number]BBB B={ Alphabet character or Arabic number}	1FA7

PubChem [8]	CCCC+[any digit number],
	CCCC={PubChem, PubChem-Substance,
	PubChem-Compound,
	PubChem-BioAssay}
		PubChem/12345,
		PubChem-Substance/
		123456

SwissProt, PIR, GDB, CSD, HGML, PREFSEQDB [9]	Free Formats	Free Formats

Table 4.1. Databank names and DAN formats.

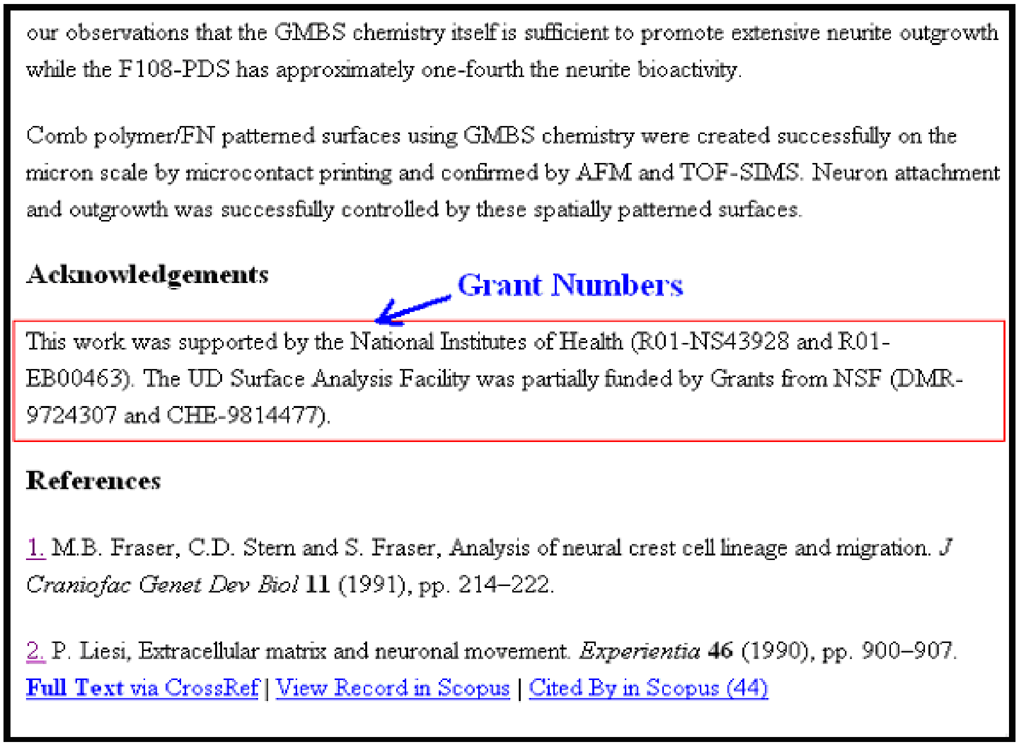
The names of databanks and formats of DANs are shown in Table 4.1. Several examples of DANs are shown in the third column.

Grant Number (GN)

A grant number [10] published in a journal article indicates a number assigned to the funding provided by the institution that supported the research reported in the article. Funding sources may be agencies of the U.S. Public Health Service (PHS), foreign governments, or private organizations such as the Wellcome Trust.

A GN in a sentence is usually accompanied by organizational names and/or words such as “supported”, “funded”, “financed”, etc. A typical example of a GN sentence is “This work was supported by National Institutes of Health Grant GM46904”. In this example, “GM46904” is the grant number in which “GM” stands for the “*National Institute of General Medical Sciences (NIGMS)*” at NIH.

Figure 4.3 shows examples of GNs in articles. Figure 4.3 (a) shows GNs R01-NS43928 and R01-EB00463 in which “NS” and “EB” stand for “*National Institute of Neurological Disorders and Stroke (NINDS)*” and “*National Institute of Biomedical Imaging and Bioengineering (NIBIB)*”, respectively. Figure 4.3 (b) shows a GN 5R01AI20451-18 in which “AI” stands for “*National Institute of Allergy and Infectious Diseases Extramural Activities (NIAID)*”. In these examples, GNs include a two-character identifier representing the name of the specific funding organization within NIH.



our observations that the GMBS chemistry itself is sufficient to promote extensive neurite outgrowth while the F108-PDS has approximately one-fourth the neurite bioactivity.

Comb polymer/FN patterned surfaces using GMBS chemistry were created successfully on the micron scale by microcontact printing and confirmed by AFM and TOF-SIMS. Neuron attachment and outgrowth was successfully controlled by these spatially patterned surfaces.

Acknowledgements

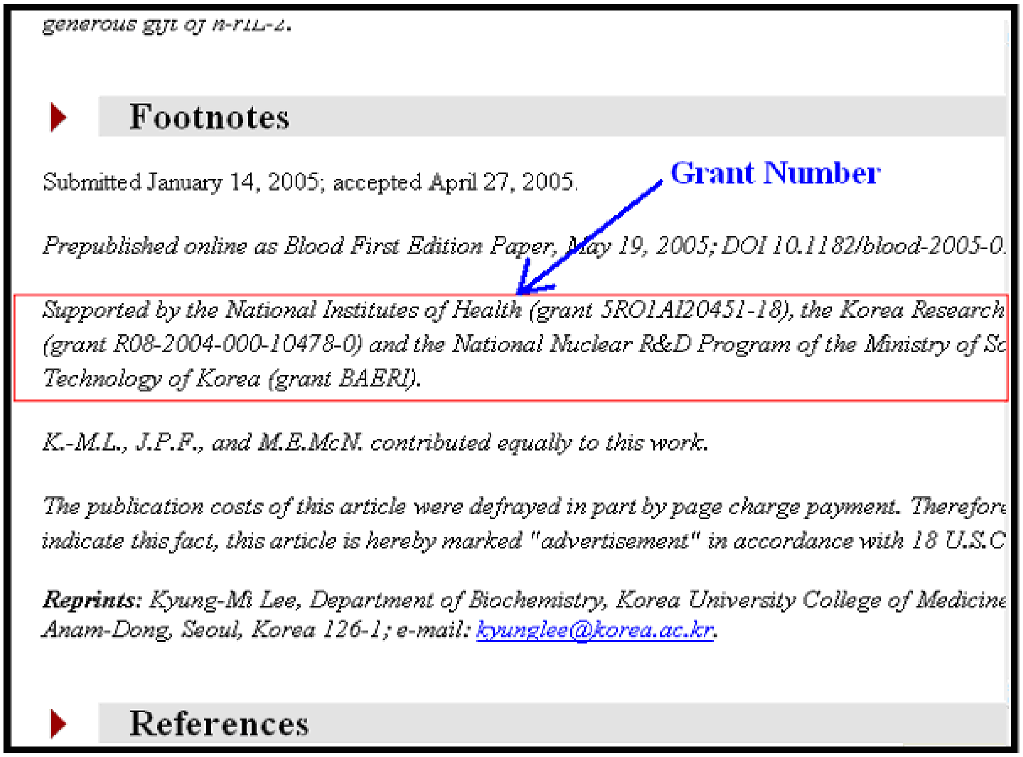
This work was supported by the National Institutes of Health (R01-NS43928 and R01-EB00463). The UD Surface Analysis Facility was partially funded by Grants from NSF (DMR-9724307 and CHE-9814477).

References

1. M.B. Fraser, C.D. Stern and S. Fraser, Analysis of neural crest cell lineage and migration. *J Craniofac Genet Dev Biol* **11** (1991), pp. 214–222.

2. P. Liesi, Extracellular matrix and neuronal movement. *Experientia* **46** (1990), pp. 900–907. Full Text via CrossRef | View Record in Scopus | Cited By in Scopus (44)

(a)



generous gift of h-rIL-2.

Footnotes

Submitted January 14, 2005; accepted April 27, 2005.

Prepublished online as Blood First Edition Paper, May 19, 2005; DOI 10.1182/blood-2005-0

Supported by the National Institutes of Health (grant 5R01AI20451-18), the Korea Research (grant R08-2004-000-10478-0) and the National Nuclear R&D Program of the Ministry of Sc Technology of Korea (grant BAERI).

K.-M.L., J.P.F., and M.E.McN. contributed equally to this work.

The publication costs of this article were defrayed in part by page charge payment. Therefore indicate this fact, this article is hereby marked "advertisement" in accordance with 18 U.S.C

Reprints: *Kyung-Mi Lee, Department of Biochemistry, Korea University College of Medicine Anam-Dong, Seoul, Korea 126-1; e-mail: kyunglee@korea.ac.kr.*

References

(b)

Figure 4.3. Examples of Grant Numbers.
In (a) GNs are R01-NS43928 and R01-EB00463. In (b) GN is 5R01AI20451-18.

NIH grant numbers follow a six-part format shown in Table 4.2, each part having a distinct meaning: Application Type (a single-digit code identifying the type of application for funding), Activity Code (a three-digit code identifying a specific category of extramural activity), Administering Organization (a two-letter code identifying the particular institute or center at NIH issuing the grant), Serial Number (a five- or six-digit identifier), Suffix Grant Year (a two-digit budget period), and Suffix Other (a four-digit code). Table 4.2 shows an example of a GN. Table 4.3 shows the detailed meaning of each part of the format of a GN and an example of each part is shown in the third column of Table 4.3.

Part	**Application Type**	**Activity Code**	**Administering Organization**	**Serial Number**	**Suffix Grant Year**	**Suffix Other**
Example	3	R01	CA	12329	04	S1A1

Table 4.2. An NIH grant number (example).

Part	**Explanation**	**Example** **3 R01 CA 12329 04S1A1**
Application Type	A single-digit code identifying the type of application received and processed.	3 (a supplemental request for additional funds)
Activity Code	A three-digit code identifying a specific category of extramural activity.	R01
Administering Organization	A two-letter code identifying the first major-level subdivision.	CA: National Cancer Institute LM: National Library of Medicine
Serial Number	A five (or six)-digit number assigned sequentially to a series with an institute, center, or division.	12329
Suffix Grant Year	A two-digit number indicates the actual segment or budget period of a project.	02 (grants in their second year)
Suffix Other	A four digit code signifying a Supplement (S), Amendment (A), or Allowance(X).	S1A1

Table 4.3. The six parts of an NIH grant number (definitions)

As mentioned already, besides NIH, grants are also made by other agencies of the U.S. Public Health Service, e.g., CDC and AHCPR, as well as private organizations. The grants made by these organizations are identified by numbers that follow different formats, all of which must be extracted by the PDR system.

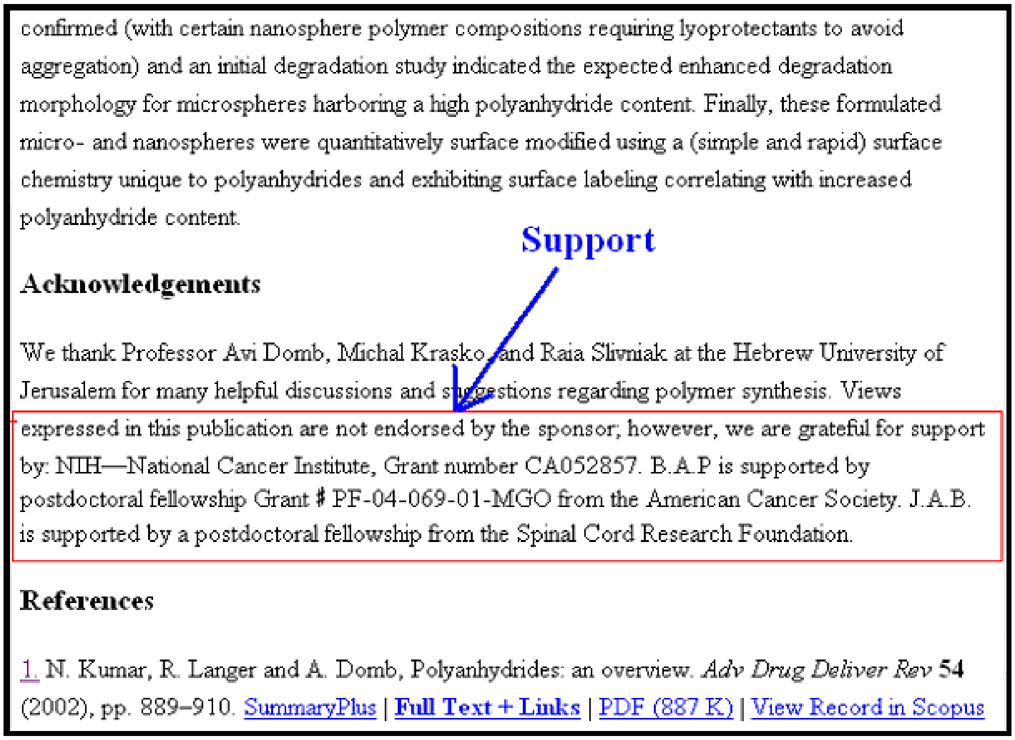
Grant Support (GS)

GS refers to the *category* of granting institution [11] as listed in Table 4.4. Authors usually acknowledge these organizations in sentences containing organizational names and “support words” such as “supported”, “funded”, “financed”, etc. A typical example of a GS sentence is “This work was supported by National Library of Medicine Grant LM46904”, where the author has identified NLM as the granting institution. PDR must use clues such as these to deduce the category of grant support, and automatically provide a check tag for operator verification.

Grant Support Category	Explanation
Support-Non US Gov	Support from universities, companies, private institutions, foreign countries, etc.
Support-US Gov Non PHS	Support from US government, other than PHS organizations.
Support-US Gov PHS	Support from one of seven PHS organizations such as NIH, FDA, HRSA, CDC, OASH, SAMHSA, and AHCPR.
Support-NIH Extramural	Support from an NIH institute or center.
Support-NIH Intramural	Support from one of the NIH organizations for intramural research.
Support-Wellcome Trust	Support from the Wellcome Trust

Table 4.4. Six categories of GSs.

Figure 4.4 shows examples of articles acknowledging the funding source. Figure 4.4 (a) shows GSs from NIH (Support-NIH Extramural), American Cancer Society (Support-Non US Gov.), and Spinal Cord Research Foundation (Support- Non US Gov). Figure 4.4 (b) shows GSs from NIH (Support-NIH Extramural).



confirmed (with certain nanosphere polymer compositions requiring lyoprotectants to avoid aggregation) and an initial degradation study indicated the expected enhanced degradation morphology for microspheres harboring a high polyanhydride content. Finally, these formulated micro- and nanospheres were quantitatively surface modified using a (simple and rapid) surface chemistry unique to polyanhydrides and exhibiting surface labeling correlating with increased polyanhydride content.

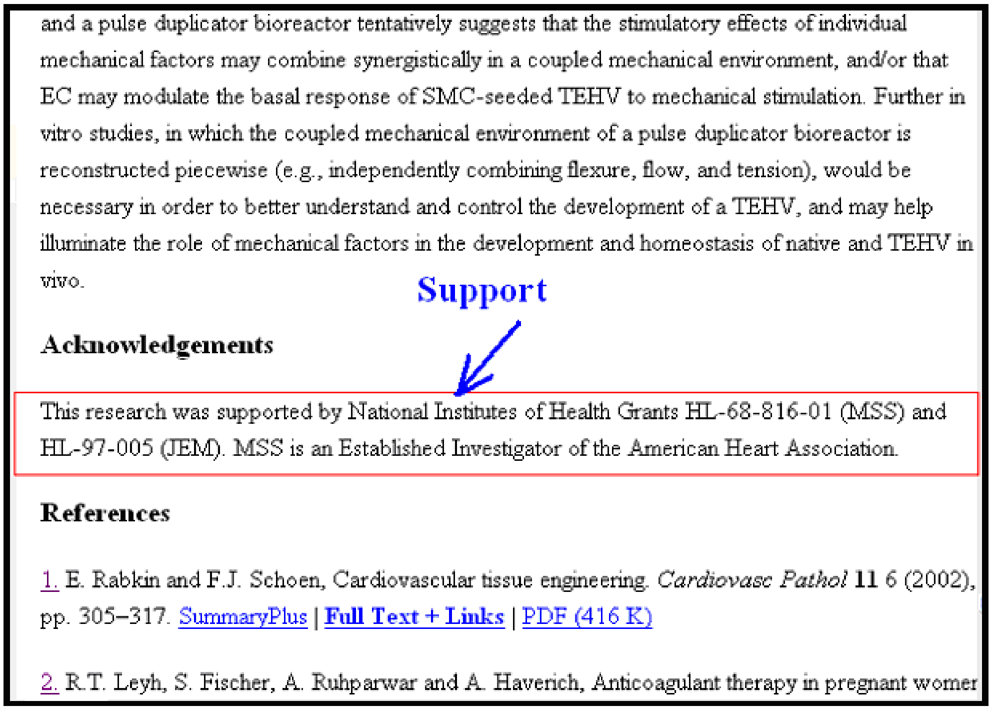
Acknowledgements

We thank Professor Avi Domb, Michal Krasko, and Raia Slivniak at the Hebrew University of Jerusalem for many helpful discussions and suggestions regarding polymer synthesis. Views expressed in this publication are not endorsed by the sponsor; however, we are grateful for support by: NIH—National Cancer Institute, Grant number CA052857. B.A.P is supported by postdoctoral fellowship Grant # PF-04-069-01-MGO from the American Cancer Society. J.A.B. is supported by a postdoctoral fellowship from the Spinal Cord Research Foundation.

References

1. N. Kumar, R. Langer and A. Domb, Polyanhydrides: an overview. *Adv Drug Deliver Rev* **54** (2002), pp. 889–910. SummaryPlus | **Full Text + Links** | PDF (887 K) | View Record in Scopus

(a)



and a pulse duplicator bioreactor tentatively suggests that the stimulatory effects of individual mechanical factors may combine synergistically in a coupled mechanical environment, and/or that EC may modulate the basal response of SMC-seeded TEHV to mechanical stimulation. Further in vitro studies, in which the coupled mechanical environment of a pulse duplicator bioreactor is reconstructed piecewise (e.g., independently combining flexure, flow, and tension), would be necessary in order to better understand and control the development of a TEHV, and may help illuminate the role of mechanical factors in the development and homeostasis of native and TEHV in vivo.

Acknowledgements

This research was supported by National Institutes of Health Grants HL-68-816-01 (MSS) and HL-97-005 (JEM). MSS is an Established Investigator of the American Heart Association.

References

1. E. Rabkin and F.J. Schoen, Cardiovascular tissue engineering. *Cardiovasc Pathol* **11** 6 (2002), pp. 305–317. SummaryPlus | **Full Text + Links** | PDF (416 K)

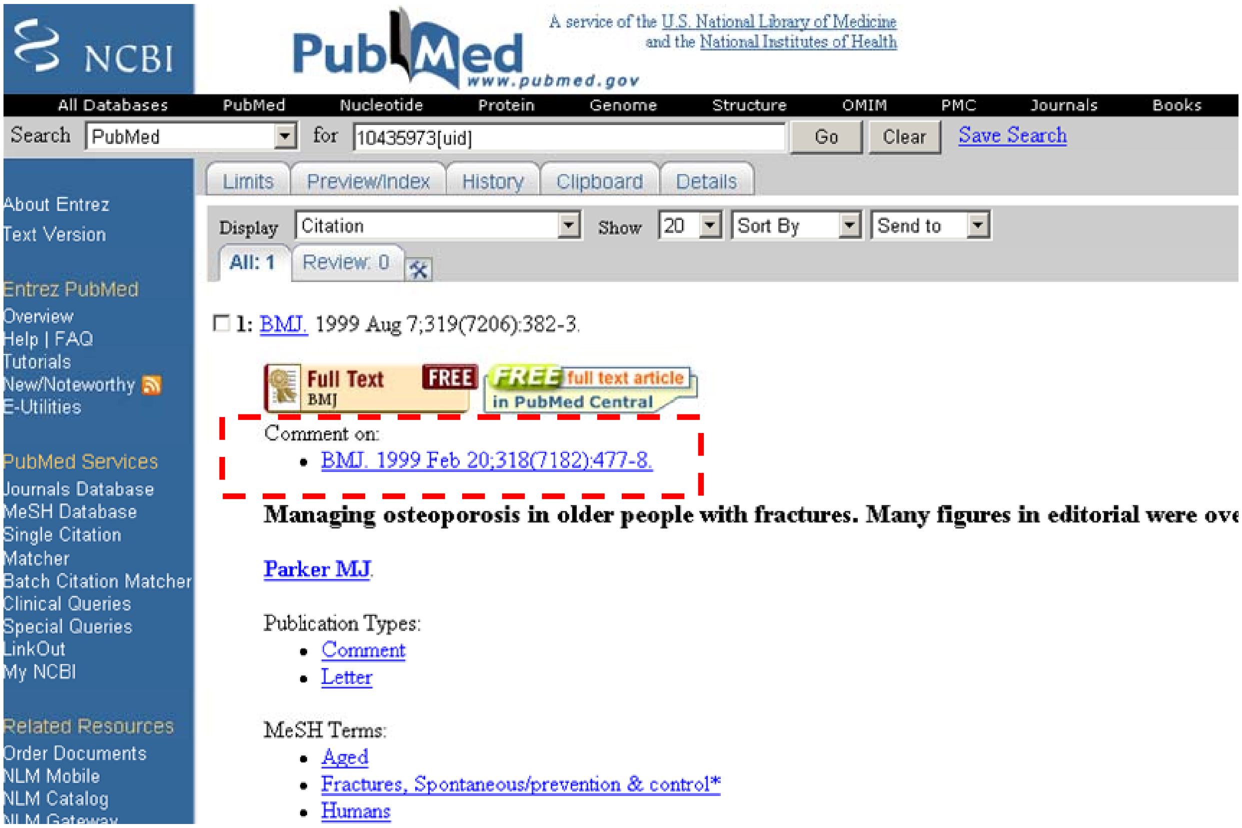
2. R.T. Leyh, S. Fischer, A. Ruhparwar and A. Haverich, Anticoagulant therapy in pregnant womer

(b)

Figure 4.4. Examples of Grant Support.
(a) GS: NIH-Extramural and Non US Gov. (b) GS: NIH-Extramural.

Commented-on Article (CON field in MEDLINE)

Increasingly, authors cite other people who have previously published articles that address related research, and they do so generally in a complimentary way. These “commented on” articles are indicated now in a MEDLINE citation field, “Comment-on” or CON. As an example, Fig. 4.5(a) is a MEDLINE citation in which this CON information is shown enclosed in a dotted box. Also, as shown in Fig. 4.5(b), an author usually provides the bibliographic description of such a commented-on article in the reference section.



Search PubMed for 10435973[uid]

Display Citation Show 20

□ 1: BMJ. 1999 Aug 7;319(7206):382-3.

Comment on:

- BMJ. 1999 Feb 20;318(7182):477-8.

Managing osteoporosis in older people with fractures. Many figures in editorial were ove

Parker MJ.

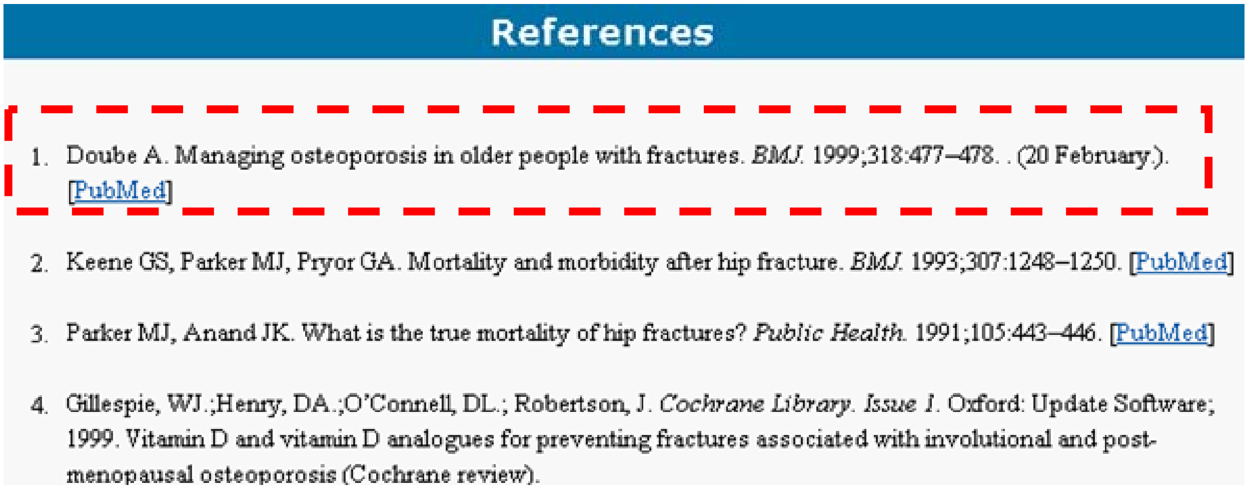
Publication Types:

- Comment
- Letter

MeSH Terms:

- Aged
- Fractures, Spontaneous/prevention & control*
- Humans

(a)



References

1. Doube A. Managing osteoporosis in older people with fractures. *BMJ.* 1999;318:477–478. . (20 February.). [PubMed]
2. Keene GS, Parker MJ, Pryor GA. Mortality and morbidity after hip fracture. *BMJ.* 1993;307:1248–1250. [PubMed]
3. Parker MJ, Anand JK. What is the true mortality of hip fractures? *Public Health.* 1991;105:443–446. [PubMed]
4. Gillespie, WJ.;Henry, DA.;O'Connell, DL.; Robertson, J. *Cochrane Library. Issue 1.* Oxford: Update Software; 1999. Vitamin D and vitamin D analogues for preventing fractures associated with involutional and post-menopausal osteoporosis (Cochrane review).

(b)

Figure 4.5. (a) A MEDLINE citation showing CON information and (b) the corresponding reference in the article text.

5. Input and output subsystems

Before describing our algorithm design in Sections 6 and 7, we first discuss the initial process of retrieving the publishers' XML citation data from NLM's DCMS system and acquiring the articles from publisher Web sites ("input"), as well as the final step of displaying the data extracted by PDR for verification by operators ("output").

5.1 Get Citations From DCMS Queue

As shown in Fig. 4.1, the input process begins with a module accessing DCMS, and retrieving the publishers' XML files (as shown in Fig. 4.1). Since a list of XML files is released daily by DCMS at 5:30pm, our module accesses DCMS at 6:30pm, and downloads any new files. These files are saved to a particular file server location, and the module then parses the files for information such as "PMID", "NLM Unique ID", "Journal Name", and "Journal Name Abbreviation," and saves this information into a database for article downloading. Other information such as Title, Abstract, Authors, Affiliations, etc. are also saved for later processing such as full-text verification and labeling improvement.

5.2 HTML/PDF Files Download

This module, also shown in Fig. 4.1, captures and saves the full text of an online article. This is done by using the article's PMID (from the previous module) in NCBI's E-utility service to get the article's URL and information on the journal and publisher. The module then loads the URL and displays the corresponding Web page in Internet Explorer. It then captures the text from the IE using detailed information from the article's Document Object Model (DOM). Since the URL provided by the E-utility sometimes links to an abstract or summary rather than to the article itself, the module further analyzes the URL and navigates to the precise location of the article. To accomplish this, the module removes <script>, <noscript> and <iframe> tags in the captured content, and compares the remaining text against the information on the article saved earlier by the **Get Citations From DCMS Queue** module (Section 5.1) to verify that the entity captured is indeed the full text of the article.

We also need to download articles without appearing to be crawling for, and illegally retrieving, articles from publishers' Web site. This could inadvertently trigger a denial of service. We comply with the publishers' recommended 10 second delay between accesses, by setting a 15 second delay if the articles are from different journal issues, and 30 seconds if they are from the same one. Furthermore, the module picks PMIDs randomly from the DCMS queue to minimize the chances of accessing consecutive articles from the same journal issue.

5.3 Text verification subsystem (Client-based PDR Reconcile)

PDR Reconcile is a graphical user interface (GUI) program to present bibliographic data to operators for verification and to transfer the verified data to DCMS. PDR Reconcile consists of two major processes, one on the client side and the other on the server side. This allows the operator to receive bibliographic data from the PDR database as shown in Figure 5.1. The client side system allows operators to verify bibliographic data received from PDR. The server side system provides internal functionalities to process requests from the client side and returns the result to the client side using XML and ASP.NET. The client side system is activated by the operator. Figure 5.2 shows an example of this system activation for reconciling DAN bibliographic data when an operator clicks on a DAN link on the DCMS user interface. Similar activation procedures are applied to review the extracted GN, GS, and CON data.

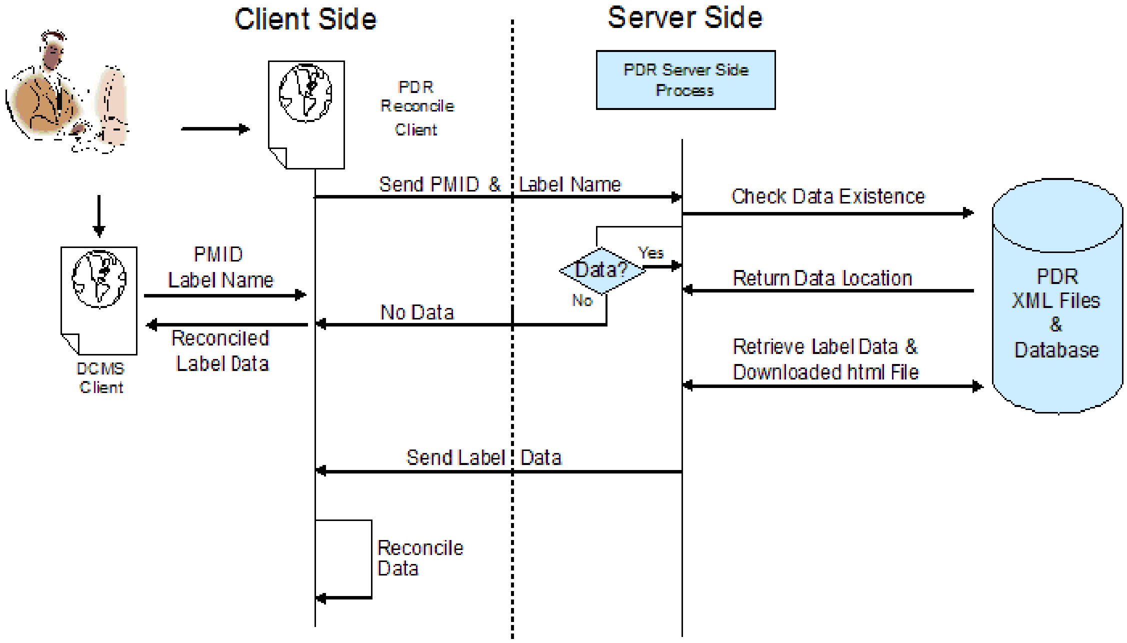
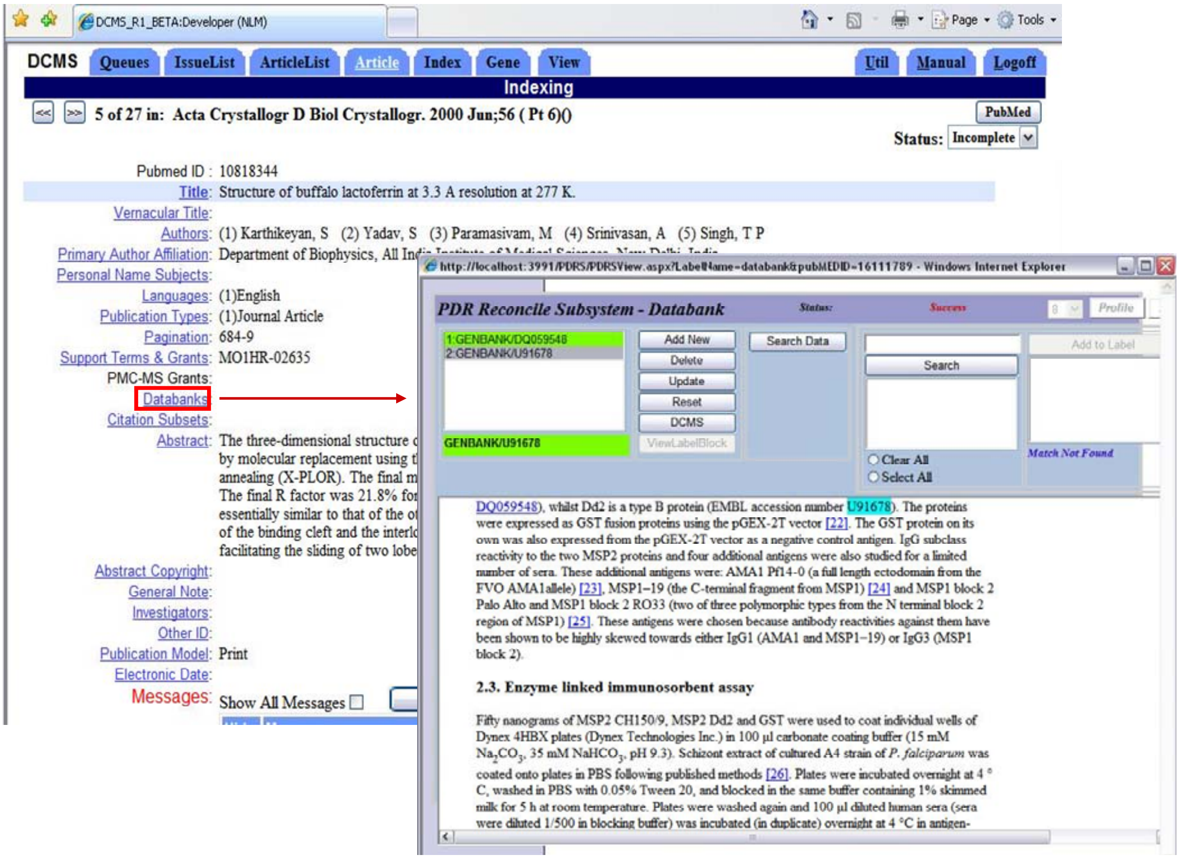


Figure 5.1. The PDR Reconcile system functional overview



5 of 27 in: Acta Crystallogr D Biol Crystallogr. 2000 Jun;56 (Pt 6)()

Status: Incomplete

Pubmed ID : 10818344
Title: Structure of buffalo lactoferrin at 3.3 A resolution at 277 K.
Vernacular Title:
Authors: (1) Karthikeyan, S (2) Yadav, S (3) Paramasivam, M (4) Srinivasan, A (5) Singh, T P
Primary Author Affiliation: Department of Biophysics, All Ind
Personal Name Subjects:
Languages: (1)English
Publication Types: (1)Journal Article
Pagination: 684-9
Support Terms & Grants: MO1HR-02635
PMC-MS Grants:
Databanks:
Citation Subsets:
Abstract: The three-dimensional structure c
by molecular replacement using t
annealing (X-PLOR). The final m
The final R factor was 21.8% for
essentially similar to that of the ot
of the binding cleft and the interlo
facilitating the sliding of two lobe
Abstract Copyright:
General Note:
Investigators:
Other ID:
Publication Model: Print
Electronic Date:
Messages: Show All Messages □

http://localhost:3991/PDRS/PDRSView.aspx?LabelName=databank&pubMEDID=16111789 - Windows Internet Explorer

PDR Reconcile Subsystem - Databank

Status: Success

1:GENBANK/DQ059548
2:GENBANK/U91678

GENBANK/U91678

Match Not Found

DQ059548), whilst Dd2 is a type B protein (EMBL accession number U91678). The proteins were expressed as GST fusion proteins using the pGEX-2T vector [22]. The GST protein on its own was also expressed from the pGEX-2T vector as a negative control antigen. IgG subclass reactivity to the two MSP2 proteins and four additional antigens were also studied for a limited number of sera. These additional antigens were: AMA1 Pf14-0 (a full length ectodomain from the FVO AMA1allele) [23], MSP1–19 (the C-terminal fragment from MSP1) [24] and MSP1 block 2 Palo Alto and MSP1 block 2 RO33 (two of three polymorphic types from the N terminal block 2 region of MSP1) [25]. These antigens were chosen because antibody reactivities against them have been shown to be highly skewed towards either IgG1 (AMA1 and MSP1–19) or IgG3 (MSP1 block 2).

2.3. Enzyme linked immunosorbent assay

Fifty nanograms of MSP2 CH150/9, MSP2 Dd2 and GST were used to coat individual wells of Dynex 4HBX plates (Dynex Technologies Inc.) in 100 µl carbonate coating buffer (15 mM Na_2CO_3, 35 mM $NaHCO_3$, pH 9.3). Schizont extract of cultured A4 strain of *P. falciparum* was coated onto plates in PBS following published methods [26]. Plates were incubated overnight at 4 ° C, washed in PBS with 0.05% Tween 20, and blocked in the same buffer containing 1% skimmed milk for 5 h at room temperature. Plates were washed again and 100 µl diluted human sera (sera were diluted 1/500 in blocking buffer) was incubated (in duplicate) overnight at 4 °C in antigen-

Figure 5.2. The DCMS interface invokes the PDR Reconcile window containing extracted data for verification (in this case, DAN.)

6. Page segmentation (Zone creation)

The purpose of the page segmentation module in PDR is to divide HTML article pages into zones of contiguous text to facilitate subsequent information extraction tasks.

6.1 Previous work in the literature

Most current Web information retrieval systems regard a Web page as the smallest indivisible unit and simply search the information in the entire Web page linearly without trying to understand the structure of the page. In the narrow domain of online HTML medical journal articles, a Web page usually consists of navigation panels, advertisements, banners, decorations, and the article itself. The article text can be logically divided into several zones, containing the title, author names, affiliations, acknowledgement, references and so on. As typically done for traditional scanned documents, layout analysis (that segments a page into zones) of the HTML pages can make subsequent information extraction processes faster and more reliable.

The most straightforward way to segment a Web page is simply to use the HTML tags as indicators. For example, Diao et al. used four types of tags, <P>, <TABLE>, <LI>/<UL> and <H1>~<H6> to detect paragraphs, tables, lists and headings [12]. Lin and Ho used only the <TABLE> tag to partition a page into several blocks [13]. Similarly, Buyukkokten et al. and Kaasinen et al. chose to use several simple tags, such as <P>, <TABLE> and <UL> to divide the Web page for subsequent conversion and summarization [14, 15]. One problem with these approaches is that the HTML syntax is very flexible, and is designed for displaying and manipulating, instead of semantically understanding, the HTML pages. Visually-similar HTML pages can therefore be implemented by completely different HTML codes. Thus, using a list of predefined HTML tags for layout analysis can produce misleading results.

Another technique, VIPS (VIsion-based Page Segmentation), analyzes the document's DOM[2] tree structure, and uses a measure called Degree of Coherence to separate or group the DOM nodes, thereby segmenting the page into zones [16].

We note that, as in scanned documents, one of the most important cues to understand the semantic organization of an online journal article is its geometric layout. Therefore, unlike most of the existing methods, which mostly depend on HTML tags or the DOM tree alone, our approach relies heavily on the geometric layout of the Web page, while exploiting aspects of other techniques.

Traditional geometric layout analysis on scanned documents has been extensively studied and documented in the literature. Most of the algorithms follow either a top-down or bottom-up approach. Top-down algorithms recursively divide a whole page into smaller zones. The process terminates when certain criteria are met. Typical top-down methods include the X-Y cut [17, 18], shape-directed-covers-based algorithms [19] and several others. Bottom-up algorithms start with the image pixels, and cluster them into connected components, then into words, lines and finally zones. Typical bottom-up methods include Docstrum [20], Block Adjacency Graph (BAG) [21], and many others. Hybrid methods combining split and merge strategies have also been proposed in [22, 23]. A review of many of these techniques appears in [24].

[2] DOM stands for Document Object Model, and is a well-defined model published by World Wide Web Consortium (W3C) for accessing and manipulating HTML documents.

Our own research has been reported in detail in [25-27], but here we summarize our zoning method in the following subsections.

6.2 Method

The DOM tree is the starting point for our HTML zoning algorithm. X-Y cut, primarily based on gaps between adjacent blocks, is a simple and efficient algorithm [17, 18]. Its major drawback is that it is sensitive to skew and noise, but unlike for scanned documents, this is not a problem for online HTML pages, in which the bounding boxes of DOM nodes are straight and clean. We, therefore, combine the DOM tree and traditional recursive X-Y cut in our zoning algorithm. HTML documents are represented by a zone tree model, which hierarchically organizes the regions of the Web page into a tree structure. The gaps between the zones and other visual cues, such as background color and font attributes (size, color and typeface), are then used to prune the zone tree and appropriately segment the HTML page.

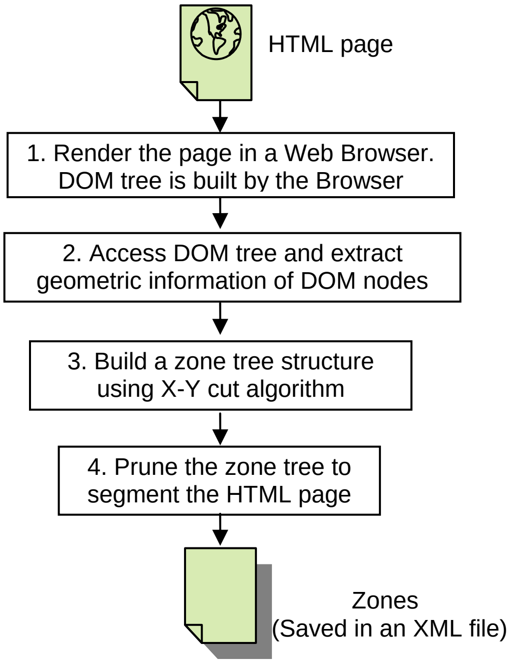
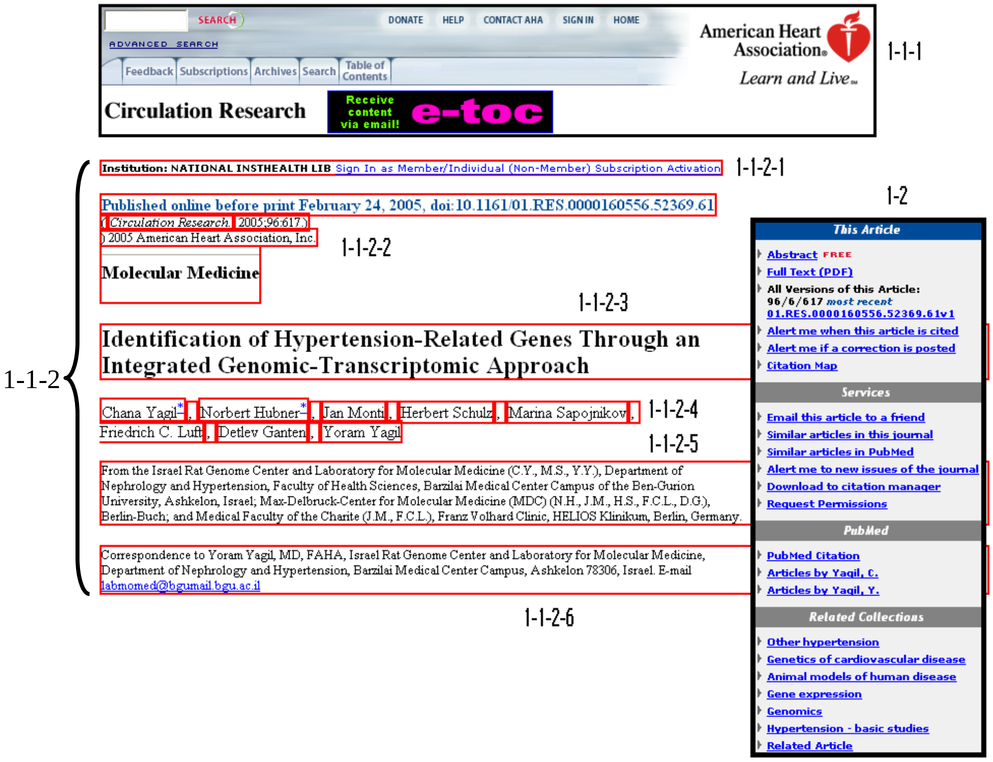

HTML page
1. Render the page in a Web Browser. DOM tree is built by the Browser
2. Access DOM tree and extract geometric information of DOM nodes
3. Build a zone tree structure using X-Y cut algorithm
4. Prune the zone tree to segment the HTML page
Zones
(Saved in an XML file)

Figure 6.1. HTML zoning algorithm

As illustrated in Figure 6.1, our zoning algorithm is a four-step process. It starts with rendering the HTML article page in a Browser. We choose to render the HTML document in Microsoft Internet Explorer (IE), because it provides simple Application Programming Interfaces (API) to create and access HTML DOM trees. Performing a preorder traversal of the DOM tree, we can readily extract geometric information (position and size of the bounding box for each DOM node) through several IE API function calls.

We then perform a recursive X-Y cut algorithm on the lowest level nodes, i.e., "leaves" of the DOM tree. This top-down process recursively breaks the page into zones based on the gaps between the bounding boxes of the DOM nodes. Visually, a zone is a region on a Web page that contains one or more DOM nodes. The root zone is the entire Web page and the DOM nodes are grouped into zones at different levels depending on their geometric relationships. This process creates a zone tree structure.



Institution: NATIONAL INSTHEALTH LIB Sign In as Member/Individual (Non-Member) Subscription Activation

Published online before print February 24, 2005, doi:10.1161/01.RES.0000160556.52369.61
(*Circulation Research*. 2005;96:617.)
) 2005 American Heart Association, Inc.

Molecular Medicine

Identification of Hypertension-Related Genes Through an Integrated Genomic-Transcriptomic Approach

Chana Yagil*, Norbert Hubner*, Jan Monti, Herbert Schulz, Marina Sapojnikov, Friedrich C. Luft, Detlev Ganten, Yoram Yagil

From the Israel Rat Genome Center and Laboratory for Molecular Medicine (C.Y., M.S., Y.Y.), Department of Nephrology and Hypertension, Faculty of Health Sciences, Barzilai Medical Center Campus of the Ben-Gurion University, Ashkelon, Israel; Max-Delbruck-Center for Molecular Medicine (MDC) (N.H., J.M., H.S., F.C.L., D.G.), Berlin-Buch; and Medical Faculty of the Charite (J.M., F.C.L.), Franz Volhard Clinic, HELIOS Klinikum, Berlin, Germany.

Correspondence to Yoram Yagil, MD, FAHA, Israel Rat Genome Center and Laboratory for Molecular Medicine, Department of Nephrology and Hypertension, Barzilai Medical Center Campus, Ashkelon 78306, Israel. E-mail labmomed@bgumail.bgu.ac.il

This Article
- Abstract FREE
- Full Text (PDF)
- All Versions of this Article: 96/6/617 *most recent* 01.RES.0000160556.52369.61v1
- Alert me when this article is cited
- Alert me if a correction is posted
- Citation Map

Services
- Email this article to a friend
- Similar articles in this journal
- Similar articles in PubMed
- Alert me to new issues of the journal
- Download to citation manager
- Request Permissions

PubMed
- PubMed Citation
- Articles by Yagil, C.
- Articles by Yagil, Y.

Related Collections
- Other hypertension
- Genetics of cardiovascular disease
- Animal models of human disease
- Gene expression
- Genomics
- Hypertension - basic studies
- Related Article

Figure 6.2. An example of zone tree structure.

Figure 6.2 shows an example of the zone tree structure in a portion of an HTML article. Several zones are marked with bounding boxes of the internal DOM nodes and labeled with numbers. The root zone, corresponding to the whole page, is broken into Zone 1-1 (not marked in the figure) and Zone 1-2. Zone 1-2 corresponds to a right-aligned <TABLE> DOM node. Such right-aligned <TABLE> nodes are usually navigation panels with no relevant information, and we choose to discard them immediately. Zone 1-1 includes the remaining components on the page. The gap between Zone 1-1-1 and Zone 1-1-2-1, being the largest, causes Zone 1-1 to be separated into Zone 1-1-1 and Zone 1-1-2. The gaps between adjacent zones from Zone 1-1-2-1 to Zone 1-1-2-6 are of the same size. Zone 1-1-2 is therefore further divided into these 6 zones.

Zones 1-2 and 1-1-1 correspond to <TABLE> DOM nodes and have sub-trees under them. Zone 1-1-2-2 is also not a leaf zone, because it actually contains line-breaks, i.e., <BR> DOM nodes. Although containing more than a dozen DOM nodes, Zone 1-1-2-4 is a leaf zone, because none of its internal DOM nodes create line breaks. Zones 1-1-2-1, 1-1-2-3, 1-1-2-5 and 1-1-2-6 are also leaf zones. DOM and zone tree structures are generally different: DOM tree models the HTML syntax, while zone tree models the geometric layout of the HTML page.

The zone tree model may be formally described as follows. The entire Web page is considered the root zone node: $Z(D)=(N,S)$. $D=\{d_i\}$ is a set of DOM nodes inside this zone. The geometric position, i.e., upper-left and lower-right rectangular boundary coordinates, of zone Z is derived from these internal DOM nodes, i.e., finding the tightest bounding box of the DOM nodes. $N = \{Z_i\}$ is a set of children zones of zone Z. $S = \{s(Z_i, Z_j) | Z_i, Z_j \in Z\}$ is a set of separators between children zones Z_i and Z_j. Recursively, child zone, $Z_i(D_i) = (N_i, S_i)$, has the same structure as Z. The leaf zone is the zone with no line-breaks inside.

For an HTML document, the zone tree generating process finds a set of appropriate separators S to partition zone Z into a set of children zones N at each level of the tree. In our implementation, the separators are the gaps between adjacent zones, with one exception: right-aligned <TABLE> nodes are immediately separated from their siblings, since, as mentioned earlier, they are usually navigation panels with no relevant information.

The most important advantage of the zone tree model is that it is independent of HTML tags. We believe that this zone tree model is better for organizing the related information within a document, and therefore better for information retrieval compared to the DOM tree model.

The last step of our zoning algorithm is to prune the zone tree to conclude the segmentation. We apply a set of rules to each zone tree node to decide whether to prune its offspring. The decision is based on the gap and the changes in appearance (whether there are significant changes in the background color, font size, font typeface, etc.) among its siblings. The leaf nodes of the pruned zone tree represent the zones in the segmented page.

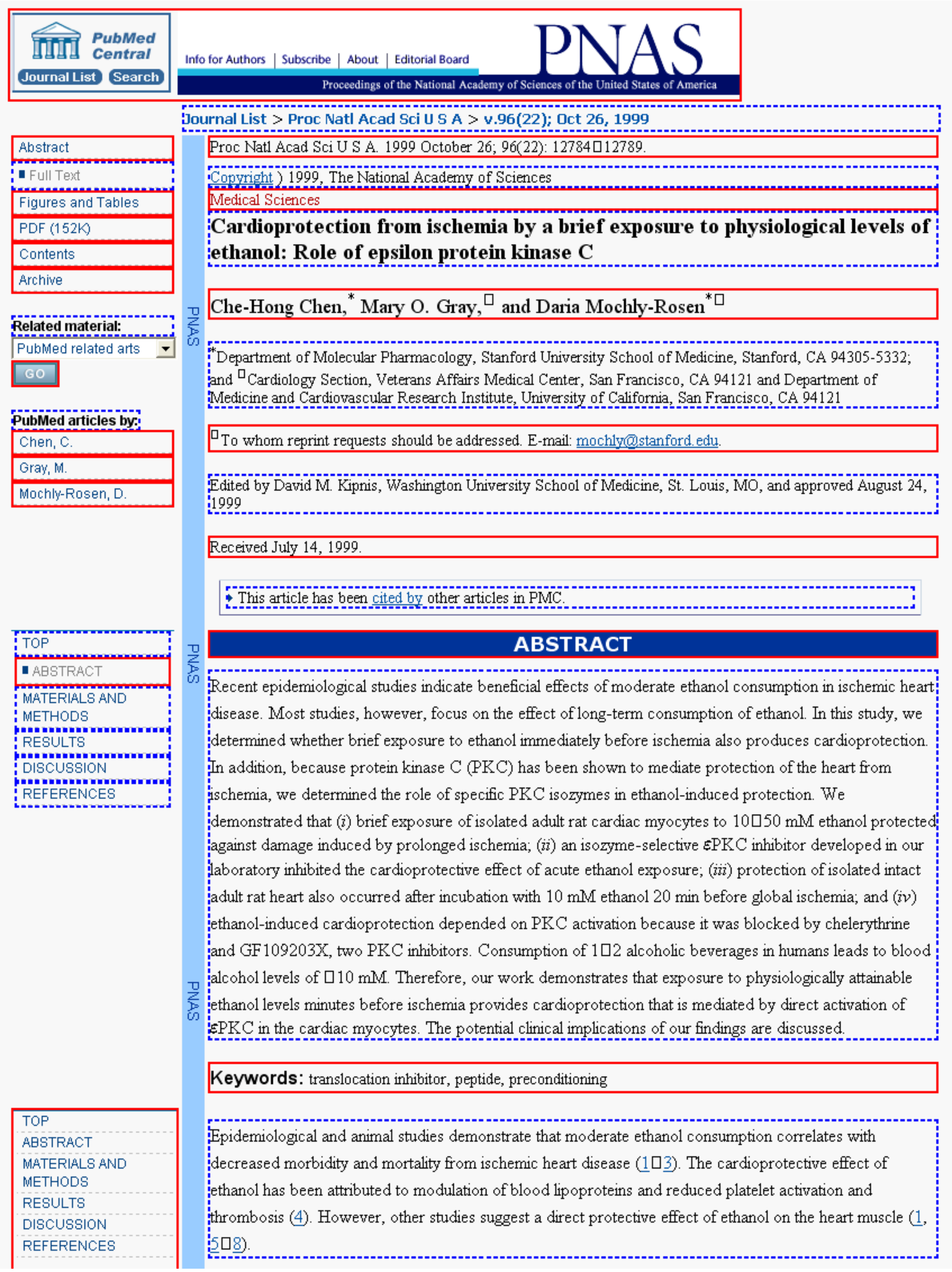
6.3 Evaluation and results

Our experimental set consists of 104 articles from 11 journals. These articles are manually segmented to provide ground truth for our evaluation. Of the 9,726 zones, the algorithm correctly identified 9,376, giving an accuracy of 96.40%.

Figures 6.3, 6.4 and 6.5 show three examples of segmentation results. The zones are indicated by solid red and dotted blue bounding boxes of internal DOM nodes. We alternate the bounding box colors to visually distinguish different zones. Note that a single zone may include several DOM nodes, and therefore several bounding boxes. The first two pages are regular HTML files, but from different publishers and implemented in different styles. The third page is a PDF-converted-to-HTML file.

Figure 6.3 shows a segmentation free of errors, which is the typical case. Errors are shown in the other two. Figure 6.4 shows a region (indicated by a parenthesis) that should be segmented into 3 zones, corresponding to author name, affiliation and Email address. Our algorithm, however, keeps the region as one zone, because they have the same font attributes, the same background color, and a small gap between them. It is difficult to separate the three without text analysis that can assign logical labels (e.g. title, author, affiliation, references, etc.) to the zones. We are, therefore, extending this research to logical layout analysis, i.e., segmenting HTML articles into zones and assigning logical labels to them.

Figure 6.5 shows an over-segmentation problem in PDF pages that are converted to HTML (marked with a thick arrow). This is a consequence of the large gaps between the keywords shown. Future work will investigate heuristic rules to correct this problem.



Proc Natl Acad Sci U S A. 1999 October 26; 96(22): 12784□12789.

Copyright) 1999, The National Academy of Sciences

Medical Sciences

Cardioprotection from ischemia by a brief exposure to physiological levels of ethanol: Role of epsilon protein kinase C

Che-Hong Chen,[*] Mary O. Gray,[□] and Daria Mochly-Rosen[*□]

[*]Department of Molecular Pharmacology, Stanford University School of Medicine, Stanford, CA 94305-5332; and [□]Cardiology Section, Veterans Affairs Medical Center, San Francisco, CA 94121 and Department of Medicine and Cardiovascular Research Institute, University of California, San Francisco, CA 94121

[□]To whom reprint requests should be addressed. E-mail: mochly@stanford.edu.

Edited by David M. Kipnis, Washington University School of Medicine, St. Louis, MO, and approved August 24, 1999

Received July 14, 1999.

- This article has been cited by other articles in PMC.

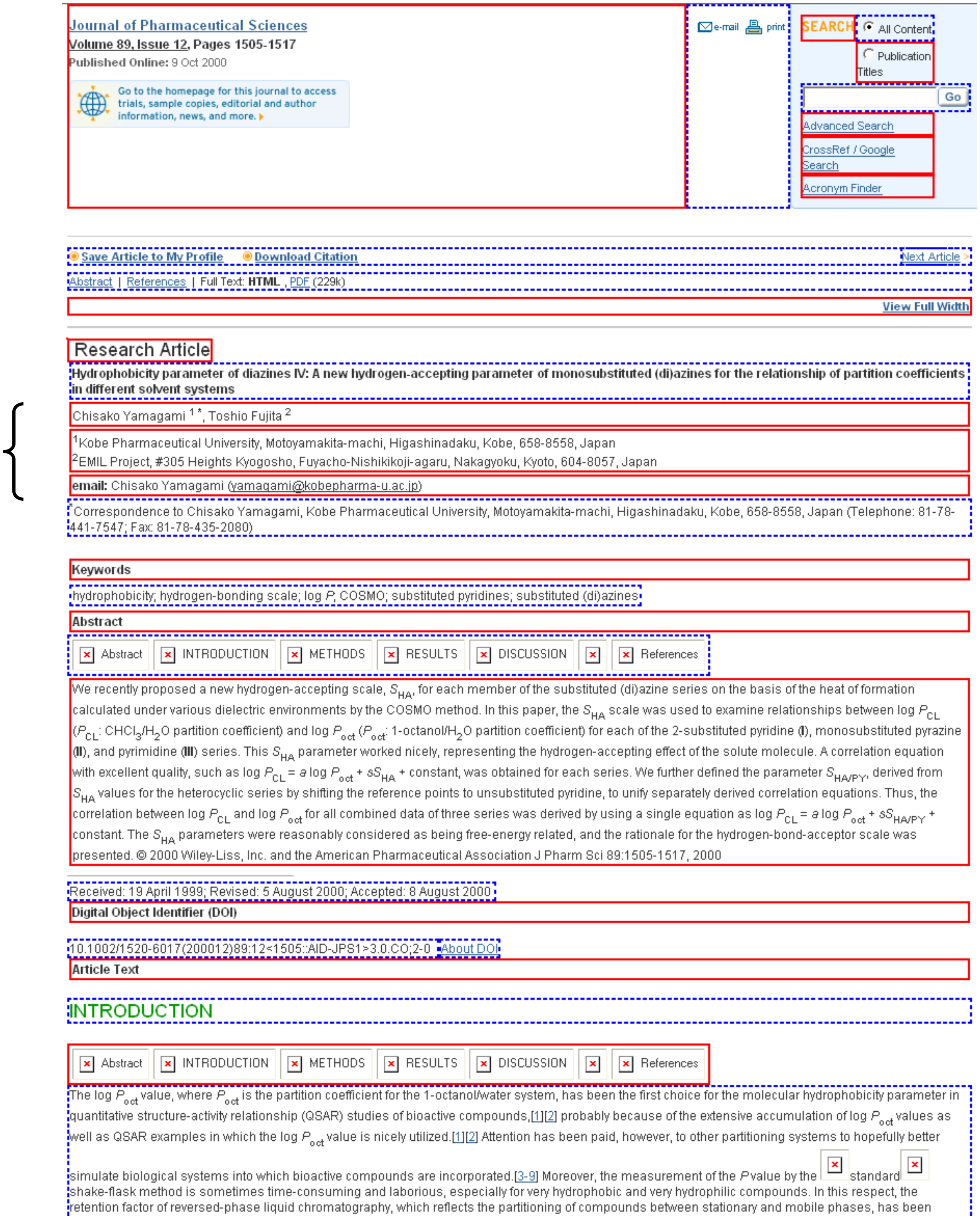
ABSTRACT

Recent epidemiological studies indicate beneficial effects of moderate ethanol consumption in ischemic heart disease. Most studies, however, focus on the effect of long-term consumption of ethanol. In this study, we determined whether brief exposure to ethanol immediately before ischemia also produces cardioprotection. In addition, because protein kinase C (PKC) has been shown to mediate protection of the heart from ischemia, we determined the role of specific PKC isozymes in ethanol-induced protection. We demonstrated that (*i*) brief exposure of isolated adult rat cardiac myocytes to 10□50 mM ethanol protected against damage induced by prolonged ischemia; (*ii*) an isozyme-selective εPKC inhibitor developed in our laboratory inhibited the cardioprotective effect of acute ethanol exposure; (*iii*) protection of isolated intact adult rat heart also occurred after incubation with 10 mM ethanol 20 min before global ischemia; and (*iv*) ethanol-induced cardioprotection depended on PKC activation because it was blocked by chelerythrine and GF109203X, two PKC inhibitors. Consumption of 1□2 alcoholic beverages in humans leads to blood alcohol levels of □10 mM. Therefore, our work demonstrates that exposure to physiologically attainable ethanol levels minutes before ischemia provides cardioprotection that is mediated by direct activation of εPKC in the cardiac myocytes. The potential clinical implications of our findings are discussed.

Keywords: translocation inhibitor, peptide, preconditioning

Epidemiological and animal studies demonstrate that moderate ethanol consumption correlates with decreased morbidity and mortality from ischemic heart disease (1□3). The cardioprotective effect of ethanol has been attributed to modulation of blood lipoproteins and reduced platelet activation and thrombosis (4). However, other studies suggest a direct protective effect of ethanol on the heart muscle (1, 5□8).

Figure 6.3. An example of error-free segmentation. Solid and dotted bounding boxes of internal DOM nodes alternate to indicate the resulting zones.



Journal of Pharmaceutical Sciences
Volume 89, Issue 12, Pages 1505-1517
Published Online: 9 Oct 2000

Go to the homepage for this journal to access trials, sample copies, editorial and author information, news, and more. ▸

Research Article

Hydrophobicity parameter of diazines IV: A new hydrogen-accepting parameter of monosubstituted (di)azines for the relationship of partition coefficients in different solvent systems

Chisako Yamagami [1] *, Toshio Fujita [2]

[1]Kobe Pharmaceutical University, Motoyamakita-machi, Higashinadaku, Kobe, 658-8558, Japan
[2]EMIL Project, #305 Heights Kyogosho, Fuyacho-Nishikikoji-agaru, Nakagyoku, Kyoto, 604-8057, Japan

email: Chisako Yamagami (yamagami@kobepharma-u.ac.jp)

*Correspondence to Chisako Yamagami, Kobe Pharmaceutical University, Motoyamakita-machi, Higashinadaku, Kobe, 658-8558, Japan (Telephone: 81-78-441-7547; Fax: 81-78-435-2080)

Keywords

hydrophobicity; hydrogen-bonding scale; log *P*; COSMO; substituted pyridines; substituted (di)azines

Abstract

☒ Abstract ☒ INTRODUCTION ☒ METHODS ☒ RESULTS ☒ DISCUSSION ☒ ☒ References

We recently proposed a new hydrogen-accepting scale, S_{HA}, for each member of the substituted (di)azine series on the basis of the heat of formation calculated under various dielectric environments by the COSMO method. In this paper, the S_{HA} scale was used to examine relationships between log P_{CL} (P_{CL}: $CHCl_3/H_2O$ partition coefficient) and log P_{oct} (P_{oct}: 1-octanol/H_2O partition coefficient) for each of the 2-substituted pyridine (**I**), monosubstituted pyrazine (**II**), and pyrimidine (**III**) series. This S_{HA} parameter worked nicely, representing the hydrogen-accepting effect of the solute molecule. A correlation equation with excellent quality, such as log P_{CL} = a log P_{oct} + sS_{HA} + constant, was obtained for each series. We further defined the parameter $S_{HA/PY}$, derived from S_{HA} values for the heterocyclic series by shifting the reference points to unsubstituted pyridine, to unify separately derived correlation equations. Thus, the correlation between log P_{CL} and log P_{oct} for all combined data of three series was derived by using a single equation as log P_{CL} = a log P_{oct} + $sS_{HA/PY}$ + constant. The S_{HA} parameters were reasonably considered as being free-energy related, and the rationale for the hydrogen-bond-acceptor scale was presented. © 2000 Wiley-Liss, Inc. and the American Pharmaceutical Association J Pharm Sci 89:1505-1517, 2000

Received: 19 April 1999; Revised: 5 August 2000; Accepted: 8 August 2000

Digital Object Identifier (DOI)

10.1002/1520-6017(200012)89:12<1505::AID-JPS1>3.0.CO;2-0 About DOI

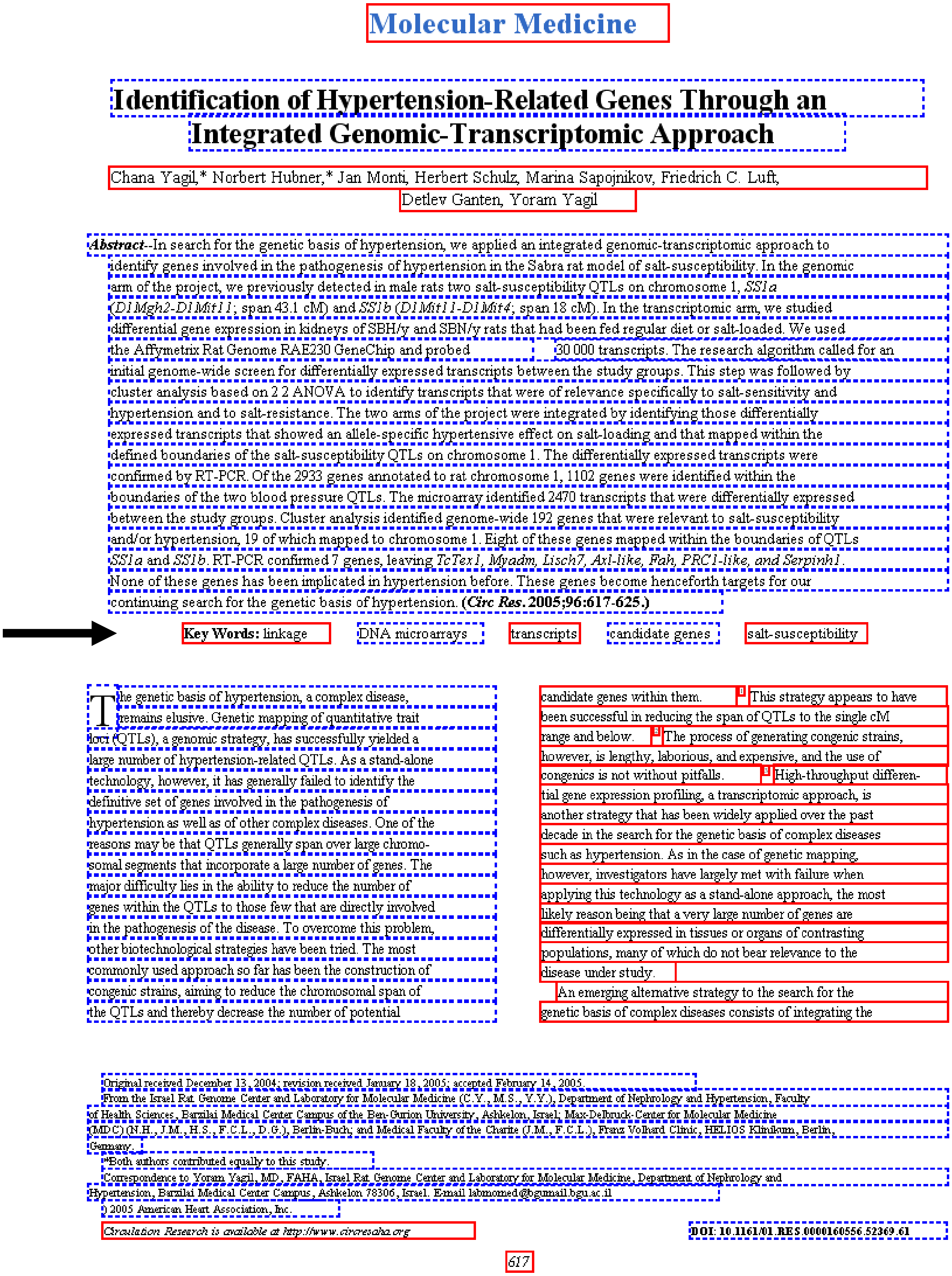
Article Text

INTRODUCTION

☒ Abstract ☒ INTRODUCTION ☒ METHODS ☒ RESULTS ☒ DISCUSSION ☒ ☒ References

The log P_{oct} value, where P_{oct} is the partition coefficient for the 1-octanol/water system, has been the first choice for the molecular hydrophobicity parameter in quantitative structure-activity relationship (QSAR) studies of bioactive compounds,[1][2] probably because of the extensive accumulation of log P_{oct} values as well as QSAR examples in which the log P_{oct} value is nicely utilized.[1][2] Attention has been paid, however, to other partitioning systems to hopefully better simulate biological systems into which bioactive compounds are incorporated.[3-9] Moreover, the measurement of the *P* value by the ☒ standard ☒ shake-flask method is sometimes time-consuming and laborious, especially for very hydrophobic and very hydrophilic compounds. In this respect, the retention factor of reversed-phase liquid chromatography, which reflects the partitioning of compounds between stationary and mobile phases, has been

Figure 6.4. An example of under-segmentation errors (shown by the parenthesis). Solid and dotted bounding boxes of internal DOM nodes alternate to indicate the resulting zones.



Molecular Medicine

Identification of Hypertension-Related Genes Through an Integrated Genomic-Transcriptomic Approach

Chana Yagil,* Norbert Hubner,* Jan Monti, Herbert Schulz, Marina Sapojnikov, Friedrich C. Luft, Detlev Ganten, Yoram Yagil

Abstract—In search for the genetic basis of hypertension, we applied an integrated genomic-transcriptomic approach to identify genes involved in the pathogenesis of hypertension in the Sabra rat model of salt-susceptibility. In the genomic arm of the project, we previously detected in male rats two salt-susceptibility QTLs on chromosome 1, *SS1a* (*D1Mgh2-D1Mit11*; span 43.1 cM) and *SS1b* (*D1Mit11-D1Mit4*; span 18 cM). In the transcriptomic arm, we studied differential gene expression in kidneys of SBH/y and SBN/y rats that had been fed regular diet or salt-loaded. We used the Affymetrix Rat Genome RAE230 GeneChip and probed 30 000 transcripts. The research algorithm called for an initial genome-wide screen for differentially expressed transcripts between the study groups. This step was followed by cluster analysis based on 2 2 ANOVA to identify transcripts that were of relevance specifically to salt-sensitivity and hypertension and to salt-resistance. The two arms of the project were integrated by identifying those differentially expressed transcripts that showed an allele-specific hypertensive effect on salt-loading and that mapped within the defined boundaries of the salt-susceptibility QTLs on chromosome 1. The differentially expressed transcripts were confirmed by RT-PCR. Of the 2933 genes annotated to rat chromosome 1, 1102 genes were identified within the boundaries of the two blood pressure QTLs. The microarray identified 2470 transcripts that were differentially expressed between the study groups. Cluster analysis identified genome-wide 192 genes that were relevant to salt-susceptibility and/or hypertension, 19 of which mapped to chromosome 1. Eight of these genes mapped within the boundaries of QTLs *SS1a* and *SS1b*. RT-PCR confirmed 7 genes, leaving *TcTex1, Myadm, Lisch7, Axl-like, Fah, PRC1-like, and Serpinh1*. None of these genes has been implicated in hypertension before. These genes become henceforth targets for our continuing search for the genetic basis of hypertension. **(*Circ Res.* 2005;96:617-625.)**

Key Words: linkage DNA microarrays transcripts candidate genes salt-susceptibility

The genetic basis of hypertension, a complex disease, remains elusive. Genetic mapping of quantitative trait loci (QTLs), a genomic strategy, has successfully yielded a large number of hypertension-related QTLs. As a stand-alone technology, however, it has generally failed to identify the definitive set of genes involved in the pathogenesis of hypertension as well as of other complex diseases. One of the reasons may be that QTLs generally span over large chromosomal segments that incorporate a large number of genes. The major difficulty lies in the ability to reduce the number of genes within the QTLs to those few that are directly involved in the pathogenesis of the disease. To overcome this problem, other biotechnological strategies have been tried. The most commonly used approach so far has been the construction of congenic strains, aiming to reduce the chromosomal span of the QTLs and thereby decrease the number of potential candidate genes within them. This strategy appears to have been successful in reducing the span of QTLs to the single cM range and below. The process of generating congenic strains, however, is lengthy, laborious, and expensive, and the use of congenics is not without pitfalls. High-throughput differential gene expression profiling, a transcriptomic approach, is another strategy that has been widely applied over the past decade in the search for the genetic basis of complex diseases such as hypertension. As in the case of genetic mapping, however, investigators have largely met with failure when applying this technology as a stand-alone approach, the most likely reason being that a very large number of genes are differentially expressed in tissues or organs of contrasting populations, many of which do not bear relevance to the disease under study.

An emerging alternative strategy to the search for the genetic basis of complex diseases consists of integrating the

Original received December 13, 2004; revision received January 18, 2005; accepted February 14, 2005.

From the Israel Rat Genome Center and Laboratory for Molecular Medicine (C.Y., M.S., Y.Y.), Department of Nephrology and Hypertension, Faculty of Health Sciences, Barzilai Medical Center Campus of the Ben-Gurion University, Ashkelon, Israel; Max-Delbruck-Center for Molecular Medicine (MDC) (N.H., J.M., H.S., F.C.L., D.G.), Berlin-Buch; and Medical Faculty of the Charite (J.M., F.C.L.), Franz Volhard Clinic, HELIOS Klinikum, Berlin, Germany.

*Both authors contributed equally to this study.

Correspondence to Yoram Yagil, MD, FAHA, Israel Rat Genome Center and Laboratory for Molecular Medicine, Department of Nephrology and Hypertension, Barzilai Medical Center Campus, Ashkelon 78306, Israel. E-mail labmomed@bgumail.bgu.ac.il

) 2005 American Heart Association, Inc.

Circulation Research is available at http://www.circresaha.org

DOI: 10.1161/01.RES.0000160556.52369.61

617

Figure 6.5. An example of over-segmentation errors (shown by the arrow).

7. Zone labeling

The step following the segmentation of article pages into zones is the labeling of those zones containing the bibliographic data of interest. We have developed two labeling techniques, one a combination of a Naïve Bayesian and a rule-based algorithm (Section 7A), and the other based on the Support Vector Machine (Section 7B). We evaluate these separately but will consider combining them eventually.

7A. Naïve Bayesian and rule-based algorithms for labeling zones

7A.1 Previous work

We earlier developed a labeling system using a rule-based algorithm (if-then-else rules) for online articles [28-31]. It works reasonably well in general cases. However, since the rules depend on combinations of keywords found in a zone, errors result when authors use unusual or ambiguous words. Consequently, the algorithm is found to be case sensitive, sensitive to typographic errors, and hence not robust.

For the labeling module in PDR we adopt the Naïve Bayesian algorithm [32, 33], commonly used in text mining and information retrieval since it is simple and efficient. It relies on the occurrence of features which are assumed to be stochastically independent of each other. Compared to our previous rule-based algorithm, it is more robust, since it can make use of any number of words in a document as features, rather than just some of the words. However, we make use of rules in our present method as well, as discussed below.

In this section, we discuss our technique for labeling text zones containing databank accession number (DAN), grant number (GN), and grant support (GS) using a combination of a Naïve Bayesian algorithm and a rule-based algorithm.

7A.2 Our approach

Since the Naïve Bayesian algorithm is based on statistics and can take advantage of several words in a zone, it is robust with respect to typographic and other errors. However, training it for a rare case is problematic. For example, the databank name "PIR" occurs very rarely and consequently there are few training examples. This can be rectified by adding if-then-else rules. Our approach therefore is to combine the Naïve Bayesian algorithm and the rule-based algorithm to exploit their relative strengths.

Figure 7.1 shows the workflow of the labeling system. The labeling system reads an XML file containing text zones in an article, labels the zones, and saves the labeling results in an XML file. Our system is a combination of three different algorithms, one each for labeling the zones as containing GN, DAN, or GS. Each one is a Naïve Bayesian algorithm supplemented by if-then-else rules.

In the case of GN, the algorithm implements the following steps. First, a zone is split into sentences since the algorithm is designed to process at the sentence level. Second, features for GN are estimated from each sentence. The algorithm collects some of the most frequently occurring words in GN training sentences as features and then checks the existence of these features in each sentence. Third, the Naïve Bayesian result for GN is computed for each sentence using the estimated features. Fourth, the label assigned to each sentence is determined by four if-then-else rules which use the following four features as their variables: the Naïve Bayesian result, Granting organization, Support word, and GN format. Fifth, the labels of all sentences in the zone are collected and saved. If one of the sentences in the zone is labeled as GN, so is the zone. Note that at the current stage of development, we have tested and implemented the Naïve Bayesian algorithms, but not the rules as yet. This will be done in the near future. The results of the evaluation of the Naïve Bayesian algorithms shown in Section 7A.3, although quite good, are expected to only improve with the addition of rules.

The labeling algorithms for DAN and GS also follow the same steps.

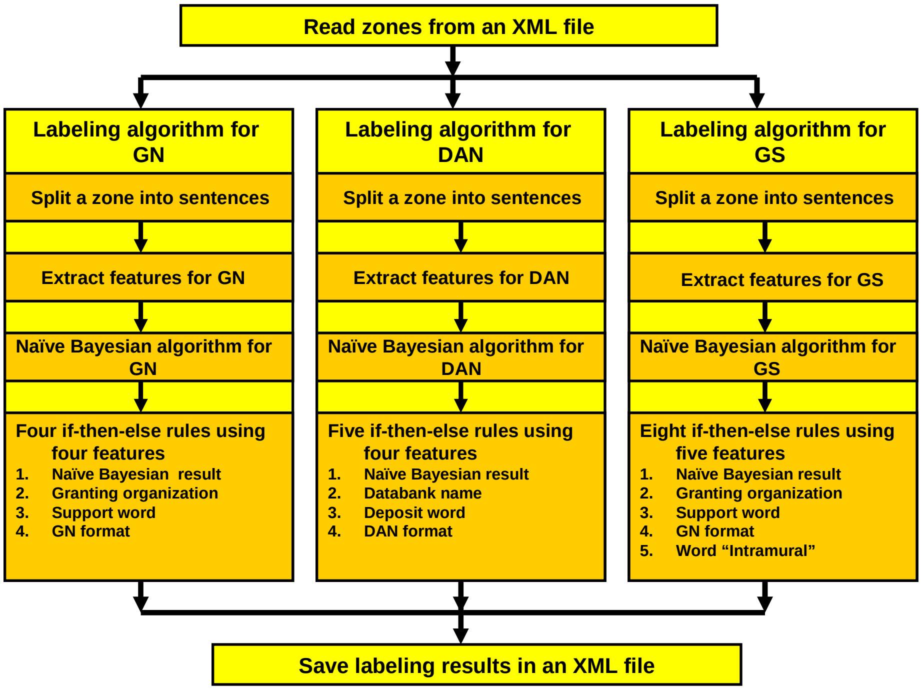

Read zones from an XML file
Labeling algorithm for GN
Split a zone into sentences
Extract features for GN
Naïve Bayesian algorithm for GN
Four if-then-else rules using four features
1. Naïve Bayesian result
2. Granting organization
3. Support word
4. GN format
Labeling algorithm for DAN
Split a zone into sentences
Extract features for DAN
Naïve Bayesian algorithm for DAN
Five if-then-else rules using four features
1. Naïve Bayesian result
2. Databank name
3. Deposit word
4. DAN format
Labeling algorithm for GS
Split a zone into sentences
Extract features for GS
Naïve Bayesian algorithm for GS
Eight if-then-else rules using five features
1. Naïve Bayesian result
2. Granting organization
3. Support word
4. GN format
5. Word "Intramural"
Save labeling results in an XML file

Figure 7.1. Labeling system.

7A.2.1 Naïve Bayesian algorithm

The design of the algorithm assumes that a document is represented by a vector of binary features indicating which words do and do not occur. I.e., assume that we have a binary feature vector from a sentence $\mathbf{x}=(x_1, x_2, x_3, \ldots, x_m)$ where m is the dimension of the vector and $x_i = 0$ or 1 means absence or presence of the ith feature (word in our case) in the sentence. The selection of the words to train the algorithm is important to successful labeling. We seek a condition or criterion to identify such words, and this is expressed in Equation (5) below.

Let us assume the existence of two classes of sentences appearing in a zone. The "relevant" class, C_r, contains words that suggest a label (e.g., GN or DAN); the other ("non-relevant") class, C_n, does not. Discrete distribution of the Bayes' Theorem may be written as

$P(C_i \mid \mathbf{x}) = \dfrac{P(\mathbf{x} \mid C_i)P(C_i)}{P(\mathbf{x})}$, i = r, n, where $P(C_i)$ is the prior probability of C_i.

The decision function can be written as

$P(\boldsymbol{x}|\mathrm{C}_r)\ P(\mathrm{C}_r) > P(\boldsymbol{x}|\mathrm{C}_n)\ P(\mathrm{C}_n)$. (1)

Assume that the features x_i in feature vector $\boldsymbol{x}=(x_1, x_2,\dots, x_m)$ are stochastically independent. Let us define p_i as the probability of occurrence of a word (suitable as a feature) in a sentence that is in the relevant class, and q_i as the probability of occurrence of such a word in the non-relevant sentence. Then, $P(\boldsymbol{x}\ |\mathrm{C}_i)$ can be rewritten as

$$P(\boldsymbol{x}\,|\,C_r) = \prod_{i=1}^{m} p_i^{x_i}(1-p_i)^{1-x_i}\text{, where } p_i = P(x_i{=}1|\ \mathrm{C}_r) \qquad (2)$$

$$P(\boldsymbol{x}\,|\,C_n) = \prod_{i=1}^{m} q_i^{x_i}(1-q_i)^{1-x_i}\text{, where } q_i = P(x_i{=}1|\mathrm{C}_n) \qquad (3)$$

Inserting Equations (2) and (3) in Equation (1), taking logs, and moving the right term to the left, we arrive at the following linear decision function $G(\boldsymbol{x})$.

$$G(\boldsymbol{x}) = \sum_{i=1}^{m}\log\frac{p_i(1-q_i)}{q_i(1-p_i)}x_i + \sum_{i=1}^{m}\log\frac{(1-p_i)}{(1-q_i)} + \log\frac{p(C_r)}{p(C_n)} \qquad (4)$$

When $G(\boldsymbol{x})$ is positive, $\boldsymbol{x}$ belongs to C_r (relevant class). If not, $\boldsymbol{x}$ belongs to C_n (non-relevant class).

To determine the selection of features, the following equation is used [34].

$$|\log\frac{p_i(1-q_i)}{q_i(1-p_i)}| >= 1 \qquad (5)$$

When a feature candidate x_i satisfies the condition expressed in Equation (5), we employ x_i as one of the features in $\boldsymbol{x}=(x_1, x_2, x_3,\dots, x_m)$. That is, this expression is the criterion we use to select the words to train the Naïve Bayesian algorithms (Section 7A.3).

7A.2.2 Rules to supplement the Naïve Bayesian algorithm

The heuristic rules we devise to supplement the Naïve Bayesian algorithm rely on key words. Table 7.1 shows the important word lists used to develop our rules.

Word list	**Words in the list**
Support word	supported, funded, granted, financed, etc.
Grant word	grant, fund, scholarship, etc.
Granting organization	NIH, FDA, CDC, , OASH, SAMHSA, AHCPR, etc.
Databank name	GenBank, Embl, Ddbj, Swiss-Prot, CSD, GDB, HGML, OMIN, PDB, PIR, PRFSEQDB
Deposit word	submit, deposit, register, etc.
Accession word	accession, access, etc.
No. word	No., Number, ID, etc.

Table 7.1. Word lists.

Rules for GN

In Table 7.2, we show rules for GN to compensate for the weakness of the Naïve Bayesian algorithm.

Rule number	**Naïve Bayesian result**	**Granting organization**	**Support word**	**GN format**
1	+	O	X	C
2	+/-	O	O	C
3	+/-	O	O	PC
4	+/-	X	O	C

Table 7.2. Rules for GN.

In this table, “+” means the result of the Naïve Bayesian algorithm is positive, “+/-” means “does not care about the result of the Naïve Bayesian algorithm”, “O” means “exist”, “X” means “does not exist”, “C” means “GN format is correct”, and “PC” means “GN format is partially correct”. These rules are made explicit below.

Rule number 1: If (a sentence has positive **Naïve Bayesian result** and
Granting organization and
correct **GN format)**,
then the sentence is labeled as GN.

Rule number 2: If (a sentence has **Granting organization** and
Support word and
correct **GN format**),
then the sentence is labeled as GN.

Rule number 3: If (a sentence has **Granting organization** and
Support Word and
partially correct **GN format**),
then the sentence is labeled as GN.

Rule number 4: If (a sentence has **Support word** and
correct **GN format**),
then the sentence is labeled as GN.

Rules for DAN

Rule generation for DAN is similar to that for GN. Five if-then-else rules are created for DAN using four features as variables: Naïve Bayesian result, Databank name, Deposit word, and DAN format. An example of the rules is given below.

Rule number 1: If (a sentence has positive **Naïve Bayesian result** and
Deposit word and
correct **DAN format**),
then the sentence is labeled as DAN.

Rules for GS

MEDLINE currently requires the identification of six types of grant support: “Non US Gov”, “US Gov Non PHS”, “US Gov PHS”, “NIH-Extramural”, “NIH-Intramural”, and “Wellcome Trust”. The rule generation for each GS is similar to that for GN. Eight if-then-else rules are created using five features as variables: Naïve Bayesian result, Support word, Grant organization, GN format, and the word “Intramural”. One of the eight rules is as follows.

Rule number 1: If (a sentence has positive **Naïve Bayesian result** and
Support Word and
correct **GN format**),
then the sentence is labeled as GS for NIH-Extramural.

When these rules are applied to a sentence, the results indicate both the existence of a GS, as well as its category (type.)

In contrast, in the case of GN and DAN, the rules only indicate the *existence* of GN and DAN in a sentence, which requires the **Bibliographic Data Extraction** subsystem in a later step to extract the specific GNs and DANs. However, in the case of GS, the rules already identify the GS category, and no further processing is needed.

7A.3. Experiment

In this section we present experimental results for labeling zones containing GN, DAN, and GS. While our module is designed to include both Naïve Bayesian as well as the rules, the results to date are only for the Naïve Bayesian algorithms, but an evaluation of the complete subsystem will be reported in the future.

7A.3.1 Naïve Bayesian algorithm

Grant number (GN)

To train the Naive Bayesian algorithm, we selected 23,500 sentences from articles cited in the MEDLINE database in 2006. Of these, 5,142 have GNs, and 18,538 do not. We also collected the 6,870 most frequently occurring words in these sentences, and select 4,721 words as features ($\boldsymbol{x}=(x_1, x_2, x_3,\ldots , x_{4,721}$)) using the criterion expressed in Equation (5).

In addition, we include three features: “Granting organization”, “Support word”, and “Grant word” for a total of 4,724 features for GN. Examples of these additional features (words) are shown in Table 7.1.

In Table 7.5 we list some of these features, and their probabilities of occurrence p_i and q_i which are derived from a frequency analysis of the training set. For example, the word “national” occurs in about 66% of the sentences in the relevant class (i.e., containing a GN), while it appears in less than 2% of the sentences in the non-relevant class, as shown in the table.

Feature	p_i	q_i
Granting organization [see Table 7.1]	0.86192143	0.01160257
Support word [see Table 7.1]	0.89478802	0.00103497
Grant word [see Table 7.1]	0.91579152	0.00076261
national	0.66297161	0.01803029
supported	0.81777518	0.00119839
grant	0.47394010	0.00054472
health	0.54453520	0.03791263
work	0.53753403	0.00114392
institutes	0.46499417	0.00544722
research	0.32360949	0.06710971
grants	0.41423571	0.00032683

Table 7.5. Some word features and corresponding p_i and q_i used in the Naïve Bayesian algorithm for GN.

The results of training and testing the Naïve Bayesian algorithm appear in Tables 7.6 and 7.7 respectively, and the corresponding performance figures are shown in Tables 7.8. All performance measures exceed 98%.

Sentence	True	False
Relevant class (W/ Grant) (Total: 5,142)	5,070	72
Non-Relevant class (W/O Grant) (Total:18,538)	74	18,284

Table 7.6. Naïve Bayesian training results for GN. (Total sentences = 23,500)

Sentence	True	False
Relevant class (W/ Grant) (Total: 5,144)	5,120	24
Non-Relevant class (W/O Grant) (Total: 18,718)	102	18616

Table 7.7. Naïve Bayesian testing results for GN. (Total sentences = 23,862)

Data Set	Precision	Recall	F-Measure
Training	98.56	99.53	99.04
Testing	98.05	99.53	98.78

Table 7.8. Performance of Naïve Bayesian results of GN.

Databank accession number (DAN)

To train the Naïve Bayesian algorithm for labeling zones containing DANs, we select 9,319 sentences from articles. 1,322 of these contain DANs and 7,997 do not. We also collect the 2,096 most frequently occurring words in these sentences and select 1,249 as features ($\boldsymbol{x}=(x_1, x_2, x_3,\ldots, x_{1,249})$) using the criterion in Equation (5). In addition, we include words from four word lists in Table 7.1: "Databank name", "Deposit word", "Accession word", and "No. word". We employ a total of 1,253 features for DAN.

Table 7.9 shows examples of these features and their corresponding p_i and q_i used in the experiments. For example, the word "accession" appears in about 63% of sentences in the relevant class, while it is virtually absent in sentences in the non-relevant class, as shown in the table.

Feature	$\boldsymbol{p_i}$	$\boldsymbol{q_i}$
Databank name [see Table 7.1]	0.93773443	0.00587647
Deposit word [see Table 7.1]	0.60465116	0.00237559
Accession word [see Table 7.1]	0.63540885	0.00012503
No. word [see Table 7.1]	0.75168792	0.14703676
Accession	0.63015754	0.00012503
Genbank	0.38484621	0.00012503
Data	0.42160540	0.02850713
Deposited	0.44786197	0.00062516
Sequence	0.28282071	0.01187797
Protein	0.28807202	0.04601150
Bank	0.25956489	0.00012503

Table 7.9. Word features and corresponding p_i and q_i used in the Naïve Bayesian algorithm for DAN.

The results of training and testing the Naïve Bayesian algorithm are shown in Tables 7.10 and 7.11, and the corresponding performance appears in Table 7.12. All three measures exceed 95.50%.

Sentence	**True**	**False**
Relevant class (W/ Databank) (Total: 1,322)	1,248	74
Non-Relevant class (W/O Databank) (Total: 7,997)	8	7,989

Table 7.10. Naïve Bayesian training result for DAN. (Total sentences = 9,319)

Sentence	**True**	**False**
Relevant class (W/ Databank) (Total: 1,312)	1,253	59
Non-Relevant class (W/O Databank) (Total: 9,302)	22	9,280

Table 7.11. Naïve Bayesian testing result for DAN. (Total sentences = 10,614)

Data Set	**Precision**	**Recall**	**F-Measure**
Training	99.36	94.40	96.81
Test	98.27	95.50	96.87

Table 7.12. Performance of Naïve Bayesian algorithm for DAN.

Grant support (GS)

From our training and testing data we observe that when a sentence has GNs, it also contains GSs. For this reason, our current method uses the Naïve Bayesian algorithm for GN to label zones containing GSs, and this is mostly successful. However, since occasionally there are sentences containing GSs with no mention of a grant number, we plan to create training and testing sets for GS as well, to improve the labeling performance.

7A.4 Summary

In summary, the performance of the Naïve Bayesian algorithm is adequate for current work. We plan to combine the algorithm with if-then-else rules to improve performance, and also include different features (such as formats of DAN and GN). We also plan to explore Decision Tree and Random Forest to generate rules automatically and to improve performance.

7B. Support Vector Machine used for labeling zones

In this section, we discuss our SVM method for labeling zones containing databank accession numbers (DAN) and grant numbers (GN). We focus on DANs in this discussion, since similar considerations apply to GNs.

One may ask why the zones containing these entities need to be labeled first. Why not extract them directly by regular expression matching since their formats are well defined? However, the problem is that many other entities can have the same formats. In the case of DANs, these would include 4-digit years and page numbers. Straightforward regular expression matching therefore would generate a large number of false positives. Errors can also arise from typographical errors or when authors do not precisely follow the required formats.

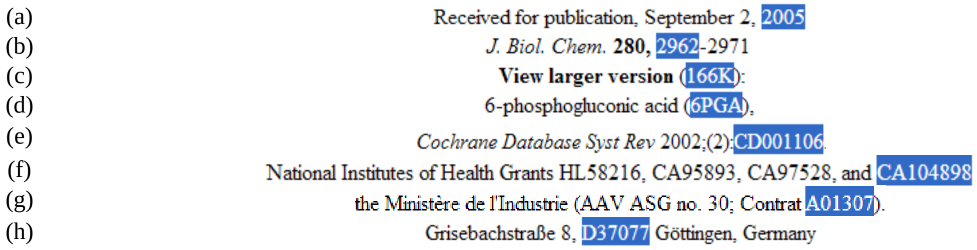
To evaluate the performance of the brute force regular expression matching approach, we conducted an experiment on 617 test articles. Out of a total of 1486 DANs, only 18 DANs are missed (false negatives), but there are 36,565 false positives. This indicates that most authors are indeed very careful about entering DANs, and due to the rigid DAN formats, the recall rate is high, 98.8%. However, because many other entities mimic DAN formats, the precision rate is very low, 3.9%. Regular expression matching, therefore, is insufficient for labeling DAN zones. Further processing is required to significantly increase the precision rate without greatly sacrificing the recall rate.

7B.1. Related work in the literature

The identification of DANs falls in the general category of *named-entity-recognition (NER)*, which typically involves the identification of locations, person names, organizations, dates, times, monetary amounts, etc., and has been well researched. In the newswire domain, the best NER algorithm can now achieve 0.95 F-score, which is considered close to human performance [35, 36].

Biomedical NER, used to identify technical terms in the biology domain (e.g. gene, protein, etc.), is of increasing interest [37]. Compared to the newswire domain, however, biomedical NER is more challenging. Several machine learning approaches have been proposed for this domain, including Support Vector Machine [38] and Conditional Random Field [39], as well as combinations of several methods to further improve performance [40].

There are two important differences between DANs and most other named entities: (1) the DANs have well-defined formats; and (2) they are sparsely located in the text. Most NER algorithms model and analyze text at the sentence level. Because DANs occur infrequently, and since most zones in an article do not contain DANs and are therefore irrelevant, it is more efficient to take a coarse-to-fine approach and conduct a zone level analysis, thereby filtering out most irrelevant zones. Then, for the remaining few candidate DAN zones, existing NER methods can be adopted to analyze sentences and extract DANs. There are two advantages of taking this coarse-to-fine approach. One is that other entities that mimic legitimate DAN formats, such as those shown in Figure 7.2, can be safely ignored, and thereby significantly increase the precision rate. The other is that due to the significant reduction in the number of candidate zones, sophisticated methods can be designed to extract poorly formatted DANs, such as those shown in Figure 7.3, and thereby increase the recall rate.



(a) Received for publication, September 2, 2005

(b) *J. Biol. Chem.* **280, 2962**-2971

(c) **View larger version** (166K):

(d) 6-phosphogluconic acid (6PGA),

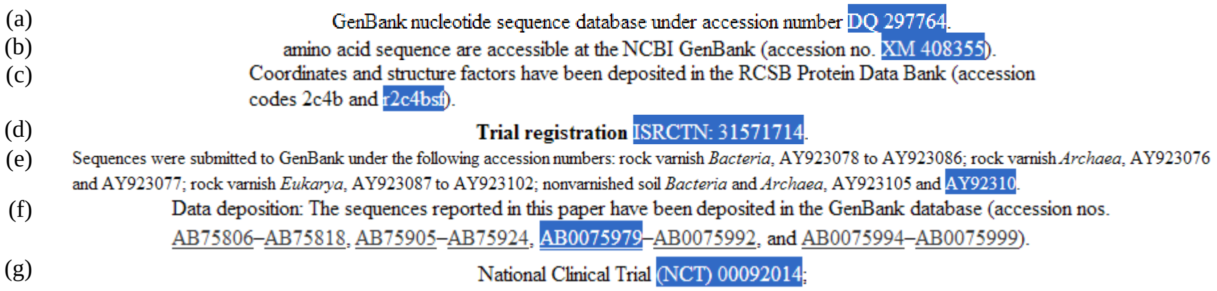
(e) *Cochrane Database Syst Rev* 2002;(2):CD001106.

(f) National Institutes of Health Grants HL58216, CA95893, CA97528, and CA104898

(g) the Ministère de l'Industrie (AAV ASG no. 30; Contrat A01307).

(h) Grisebachstraße 8, D37077 Göttingen, Germany

Figure 7.2: Examples of other entities mimicking legitimate DAN formats. Mistaken for a legitimate PDB number are (a) 4-digit year; (b) 4-digit page number; (c) file size description; (d) chemical term. Mistaken for a legitimate GenBank number are: (e) page number; (f) grant number; (g) foreign contract number; (h) foreign zip code.



(a) GenBank nucleotide sequence database under accession number DQ 297764.

(b) amino acid sequence are accessible at the NCBI GenBank (accession no. XM 408355).

(c) Coordinates and structure factors have been deposited in the RCSB Protein Data Bank (accession codes 2c4b and r2c4bsf).

(d) **Trial registration** ISRCTN: 31571714.

(e) Sequences were submitted to GenBank under the following accession numbers: rock varnish *Bacteria*, AY923078 to AY923086; rock varnish *Archaea*, AY923076 and AY923077; rock varnish *Eukarya*, AY923087 to AY923102; nonvarnished soil *Bacteria* and *Archaea*, AY923105 and AY92310.

(f) Data deposition: The sequences reported in this paper have been deposited in the GenBank database (accession nos. AB75806–AB75818, AB75905–AB75924, AB0075979–AB0075992, and AB0075994–AB0075999).

(g) National Clinical Trial (NCT) 00092014;

Figure 7.3: Examples of poorly-formatted databank accession numbers. (a) extra space between prefix and number; (b) "_" character replaced by space; (c) non-fully compatible format; (d) extra colon and space; (e) missing a digit (incorrect format); (f) extra "0"s; (g) extra parentheses and space.

Here we discuss our method for DAN zone labeling which we view as a text categorization problem. Machine learning approaches for text categorization have been intensively studied for more than a decade, particularly the Support Vector Machine [41] and boosting-based classifier committees [42]. We chose SVM for our labeling method.

SVM was originally introduced as a supervised learning algorithm for solving two-class classification problems, though it can be easily extended to handle multi-class classifications [43, 44]. Owing to its superior generalization performance, SVM has been widely used in many pattern recognition applications such as handwriting recognition [45], face detection [46], and text categorization [47].

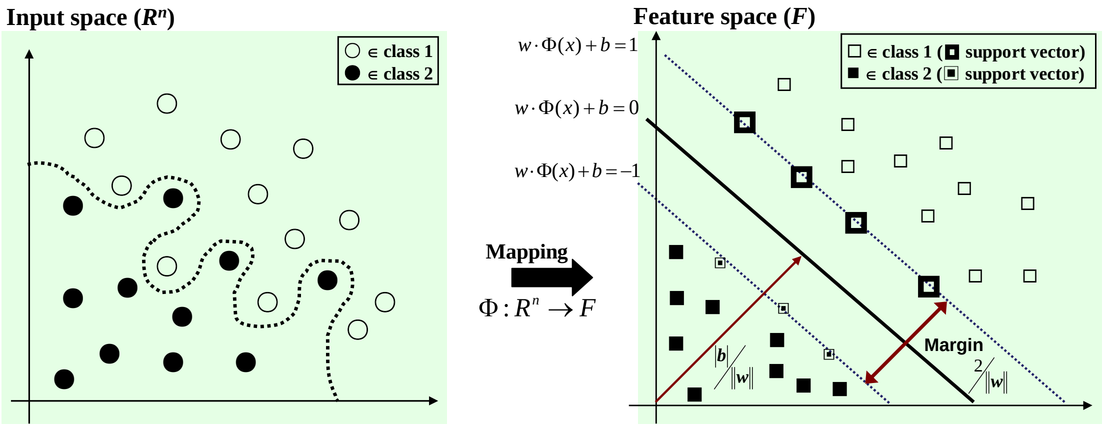
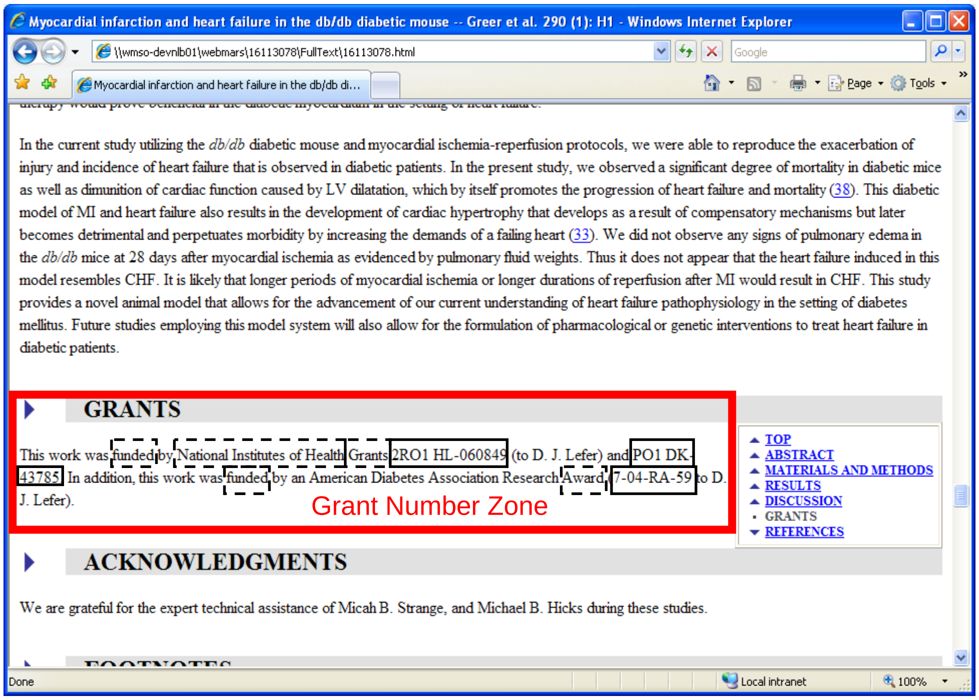

Input space (R^n)
○ ∈ class 1
● ∈ class 2
Feature space (F)
□ ∈ class 1 (▣ support vector)
■ ∈ class 2 (▣ support vector)
$w \cdot \Phi(x) + b = 1$
$w \cdot \Phi(x) + b = 0$
$w \cdot \Phi(x) + b = -1$
Mapping
$\Phi : R^n \rightarrow F$
$|b| / \|w\|$
Margin
$2 / \|w\|$

Figure 7.4. SVM learning algorithm for nonlinear separable case.

The basic idea of SVM to solve a non-linear classification problem is to map a non-linear separable input space to a linear separable higher dimensional feature space, using a predefined non-linear kernel function, and to find the optimal hyperplane that maximizes the margins between the classes in that feature space, as shown in Figure 7.4.

Grant numbers (GN) are also sparsely located in articles and are usually surrounded by distinct text as are DANs. Our hierarchical coarse-to-fine method is also applicable here. Figure 7.5 shows a GN zone marked with a thick red bounding box. Three grant numbers are highlighted with solid boxes, and the informative words, which are helpful for GN zone detection, are in the dotted boxes. Compared to DAN zones, the text inside GN zones is more consistent and distinctive, making the labeling of GN zones an easier task. In the following discussion, we use DAN zone labeling to describe our algorithm, but the same considerations hold for GN zone labeling as well.


Myocardial infarction and heart failure in the db/db diabetic mouse -- Greer et al. 290 (1): H1 - Windows Internet Explorer
\\wmso-devnlb01\webmars\16113078\FullText\16113078.html
Google
Myocardial infarction and heart failure in the db/db di...
Page
Tools
Grant Number Zone
TOP
ABSTRACT
MATERIALS AND METHODS
RESULTS
DISCUSSION
GRANTS
REFERENCES
Done
Local intranet
100%

therapy would prove beneficial in the diabetic myocardium in the setting of heart failure.

In the current study utilizing the *db/db* diabetic mouse and myocardial ischemia-reperfusion protocols, we were able to reproduce the exacerbation of injury and incidence of heart failure that is observed in diabetic patients. In the present study, we observed a significant degree of mortality in diabetic mice as well as dimunition of cardiac function caused by LV dilatation, which by itself promotes the progression of heart failure and mortality (38). This diabetic model of MI and heart failure also results in the development of cardiac hypertrophy that develops as a result of compensatory mechanisms but later becomes detrimental and perpetuates morbidity by increasing the demands of a failing heart (33). We did not observe any signs of pulmonary edema in the *db/db* mice at 28 days after myocardial ischemia as evidenced by pulmonary fluid weights. Thus it does not appear that the heart failure induced in this model resembles CHF. It is likely that longer periods of myocardial ischemia or longer durations of reperfusion after MI would result in CHF. This study provides a novel animal model that allows for the advancement of our current understanding of heart failure pathophysiology in the setting of diabetes mellitus. Future studies employing this model system will also allow for the formulation of pharmacological or genetic interventions to treat heart failure in diabetic patients.

GRANTS

This work was funded by National Institutes of Health Grants 2RO1 HL-060849 (to D. J. Lefer) and PO1 DK-43785. In addition, this work was funded by an American Diabetes Association Research Award (7-04-RA-59 to D. J. Lefer).

ACKNOWLEDGMENTS

We are grateful for the expert technical assistance of Micah B. Strange, and Michael B. Hicks during these studies.

FOOTNOTES

Figure 7.5. An example of GN zone labeling.

7B.2 Method

As shown in Figure 7.6, our SVM-based zone labeling algorithm is a two-step process. First, it extracts word frequency features from the zone. These word frequency features are concise representation of the zones, so that in the second step, the statistical SVM classifier can be trained and then used to classify unknown zones. We discuss the algorithm details below.

As mentioned, labeling DAN zones may be considered a text categorization problem, i.e., classifying zones into *DAN zones* (the zones containing DANs) and *other zones* (those without DANs).

The first step in classifying a given zone is to extract useful features to represent it, such as word frequency counts. An important question is how to choose the dictionary, i.e., the set of words to be counted. After removing stop words and rarely-appearing (less than 10 counts) words, 23,202 distinct words are collected from our training articles. It is well-known in text categorization that the high dimensionality of the word space, i.e., the large size of the dictionary, may lead to poor performance due to the so-called "curse of dimensionality." To avoid this, a dimension reduction method is employed to select an optimal word dictionary.

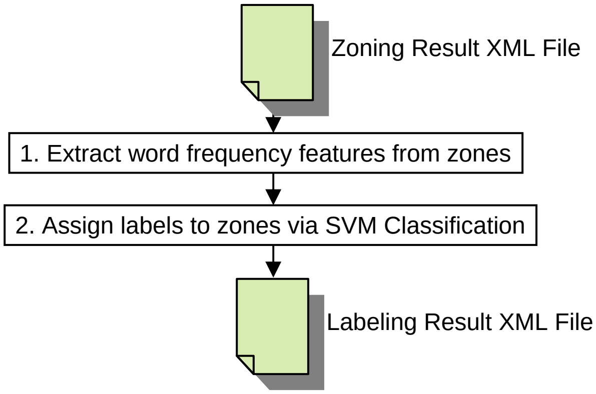

Zoning Result XML File
1. Extract word frequency features from zones
2. Assign labels to zones via SVM Classification
Labeling Result XML File

Figure 7.6. SVM zone labeling algorithm

In a survey of text categorization by Sebastiani [48], the GSS measure [49] is recognized as one of the best methods for feature dimension reduction ("GSS" named after the three authors of the referenced papers). For our classification process, GSS measures of word t_k for DAN zone label c_0 and "other zone" label c_1, are defined as:

$$GSS(t_k, c_0) = P(t_k, c_0)P(\bar{t}_k, c_1) - P(t_k, c_1)P(\bar{t}_k, c_0)$$

$$GSS(t_k, c_1) = P(t_k, c_1)P(\bar{t}_k, c_0) - P(t_k, c_0)P(\bar{t}_k, c_1)$$

where, $P(\bar{t}_k, c_i)$ indicates the probability that, given a random zone, word t_k does not appear in the zone, and that the zone belongs to category c_i. For our two-class classification, we define a joint GSS measure for each word t_k:

$$GSS(t_k) = \left| P(t_k, c_1)P(\bar{t}_k, c_0) - P(t_k, c_0)P(\bar{t}_k, c_1) \right| .$$

The GSS measure reflects the intuition that the best words are the ones distributed most differently in the DAN and other zones. $P(t_k, c_i)$ and $P(\bar{t}_k, c_i)$ can be estimated by counting occurrences in the training samples.

The 23,202 words in our training data can then be sorted according to their GSS measures. A higher value for this measure generally indicates better discriminant ability. Table 7.13 shows the 20 words with the highest GSS measures in our sample. It is of interest to find out how many words, i.e., the dictionary size, are required to achieve good performance in our DAN zone classification. An empirical study is described in Section 7B.3.4 to answer this question.

accession	deposited	genbank	data	bank	sequence	nucleotide	coordinates	sequences	abstract
text	database	numbers	protein	code	number	reported	crossref	atomic	paper

Table 7.13. The top 20 words with the highest GSS measures.

Once the word dictionary is selected, the occurrences of these words in the zone are counted. These counts form a word-frequency feature vector, denoted as $\mathbf{f_i} = \{f(t_1, d_i), \cdots, f(t_k, d_i), \cdots, f(t_n, d_i)\}$, where t_k is the kth word in the dictionary, n is the dictionary size, d_i is a zone, and $f(t_k, d_i)$ is the number of occurrences of word t_k in zone d_i. In order to make zones of different sizes comparable, the word-frequency feature vector is normalized by the total number of words in the zone. These normalized feature vectors serve to represent the zones, and are used to train the SVM classifier which then predicts the labels of the test zones.

A well-known problem with SVM classifiers is that they are sensitive to unbalanced training samples, and are biased toward the class label with more training samples. This is a problem for us since in a typical journal article, there are far more “other zones” than DAN zones, resulting in a significantly greater number of training samples for the former. We address this in an empirical study presented in Section 7B.3.3 to find the best combination of DAN and “other” zone training samples.

7B.3 Experimental evaluation

7B.3.1 Experimental data

We searched through the MEDLINE 2006 database (citations of articles indexed in 2006) to collect 1617 articles containing DANs. 1000 of them were randomly selected as training samples, and the remaining 617 as test samples. All articles were segmented into zones by our HTML segmentation algorithm briefly described in Section 6 and in detail in our published papers [25-27]. Through simple string matching, the zones containing DANs were extracted and labeled as DAN zones. The remaining ones were labeled as “other zones”. Table 7.14 summarizes the statistics of the experimental data used to evaluate our labeling algorithm.

	Articles	DANs	DAN Zones	Other Zones
Training	1000	3076	1491	66,458
Testing	617	1468	877	41,419

Table 7.14. Experimental data

7B.3.2 SVM classifier

We use LibSVM [50], an open source SVM library developed at National Taiwan University, to implement our DAN zone classification. We adopted Radial Basis Function (RBF) as the kernel function, and selected the two parameters, C (penalty parameter of the errors) and γ (RBF parameter) through an exhaustive grid-search using cross-validation on training samples. The features for representing zone text are the word frequency counts. The DAN zone classification is a fast process, taking a few hundred milliseconds to process a typical article on a conventional 3.40GHz PC equipped with 1GB RAM.

7B.3.3 Effect of unbalanced training samples

As previously mentioned, it is a known problem for SVM classifiers that unbalanced training data can seriously degrade classification performance. In our collection, there are significantly more “other zones” than DAN zones. We conducted an experiment to test how unbalanced training samples affect the classification. The 100 words with the highest GSS measures are used as the dictionary for estimating word occurrence. The evaluation is on a total of 1877 test zones, 877 of them DAN zones, and the other 1000 randomly selected from the 41,419 “other zones”. There are a total of 1491 training DAN zones, all used as training samples in our experiments. In addition, we include 372, 745, 1491, 2982 and 5964 randomly selected “other zones” into the training set. These are chosen because they are respectively ¼, ½, 1, 2, 4 times the number of training DAN zones, i.e., 1491. Table 7.15 shows the false positive (“other zones” mislabeled as DAN zones), false negative (DAN zones mislabeled as “other zones”), and average accuracies.

	False Positive	False Negative	Average Accuracy
1491 DAN and 372 other zones	19.9%	5.1%	87.5%
1491 DAN and 745 other zones	**7.1%**	**8.3%**	**92.3%**
1491 DAN and 1491 other zones	3.7%	10.4%	92.9%
1491 DAN and 2982 other zones	1.8%	13.0%	92.6%
1491 DAN and 5964 other zones	1.4%	15.2%	91.7%

Table 7.15. Results of varying the number of training samples.

This experiment clearly demonstrates that the SVM classifier is biased toward the class label with more training samples, and that classification performance is not always improved by adding more training samples. Training samples need to be balanced. Since in our problem, false negative errors (under labeling) are considered much more serious than false positive (over labeling) errors, we choose 1491 DAN and 745 “other zones” as an optimum combination to train the SVM classifier.

7B.3.4 Effect of dictionary size

Although pointed out by Sebastiani [48] that feature reduction is usually required in machine learning-based text categorization, several researchers have also shown that SVM classifiers are capable of effectively processing feature vectors of more than 10,000 dimensions [51-53]. The feature dimension, i.e., the word dictionary size, however affects not only classification accuracy, but also the computation time, which is another critical consideration in our operational system. Therefore, it is of interest to find out how dictionary size affects the performance of our DAN zone classification. Table 7.16 shows the results of varying the dictionary size from 50 to 6400.

Dictionary Size	False Positive	False Negative	Average Accuracy
50	4.7%	10.3%	92.5%
100	7.1%	8.3%	92.3%
200	9.8%	7.1%	91.5%
400	**10.7%**	**6.4%**	**91.4%**
800	12.9%	6.6%	90.2%
1600	12.7%	7.2%	90.0%
3200	11.5%	7.0%	90.7%
6400	11.6%	7.4%	90.5%

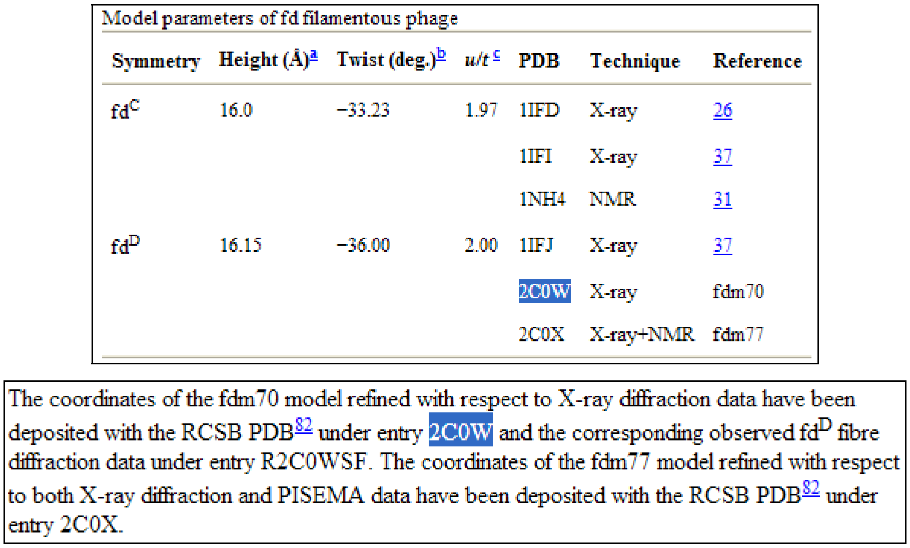
Table 7.16. Results of varying dictionary size.

SVM, as found by other researchers, is indeed robust with respect to high dimensional feature vectors. The performance drops only slightly even with a dictionary size 128 times larger. Again, since we want fewer false negatives, we select a dictionary size of 400 to give an average accuracy of 91.4%. This relatively low dimensionality also renders the SVM classifier computationally efficient.

7B.3.5 Labeling performance

As shown in Table 7.16 (bold line), in the evaluation on a test set of 1000 "other zones", 107 are misclassified as DAN zones, giving a false positive rate of 10.7%. Zones other than these 107 contain entities mimicking DAN formats. When we discard these, the precision is increased. On the test set of 877 DAN zones, 56 are missed, giving a false negative rate of 6.4%. Worth mentioning here is that the DANs are often mentioned in several places in the article. Figure 7.7 shows an example where the same DAN, "2C0W", is mentioned twice in the article, the top zone being classified as an "other zone". However, this is not a catastrophe, since the DAN may be extracted from the bottom zone. This is not a rare case. The 6.4% false negative rate in our DAN zone classification, therefore, does not mean that 6.4% DANs will be missed by the algorithm. Our final DAN extraction from labeled DAN zones is presented and evaluated in Section 8.

Compared to DAN, the text inside grant number zones is more consistent and distinctive, as mentioned earlier. GN zone labeling is therefore an easier task, and we achieved much better performance. In an evaluation on a set of 1224 test GN zones, 1220 are correctly identified, giving an accuracy of 99.7%. Usually, there is only one GN zone in an article, so we expect about 3 cases of under-labeling in every 1000 articles. Out of 1000 test "other zones", 999 are correctly labeled, for an accuracy of 99.9%. Because in a typical article there is an average of 65 "other zones", we expect one over-labeling error in every 15 articles. Our final GN extraction from labeled GN zones is also presented and evaluated in Section 8.



Model parameters of fd filamentous phage

Symmetry	Height (Å)[a]	Twist (deg.)[b]	*u*/*t* [c]	PDB	Technique	Reference
fd^{C}	16.0	−33.23	1.97	1IFD	X-ray	26
				1IFI	X-ray	37
				1NH4	NMR	31
fd^{D}	16.15	−36.00	2.00	1IFJ	X-ray	37
				2C0W	X-ray	fdm70
				2C0X	X-ray+NMR	fdm77

The coordinates of the fdm70 model refined with respect to X-ray diffraction data have been deposited with the RCSB PDB[82] under entry 2C0W and the corresponding observed fd^{D} fibre diffraction data under entry R2C0WSF. The coordinates of the fdm77 model refined with respect to both X-ray diffraction and PISEMA data have been deposited with the RCSB PDB[82] under entry 2C0X.

Figure 7.7: Databank accession numbers may appear in several places in an article. In the top, the PDB databank number 2C0W is listed in a table, which is classified as an "other zone" due to the lack of contextual information. In the bottom, the same DAN is mentioned again in a paragraph, which is classified as a DAN zone, and is therefore correctly labeled.

8. Extraction of key bibliographic data

The step following the labeling of zones in a segmented page is to automatically recognize and extract databank accession numbers (DANs), grant numbers (GNs), and articles commented on by authors (CON) from the labeled zones. In addition, this step forwards information on grant support (GS types and their confidence scores), provided by the preceding step (zone labeling) to the PDR reconcile subsystem. We describe two techniques here: in Section 8A, a hybrid contextual and statistical method to identify GNs and DANs, and in Section 8B, Support Vector Machine to identify CON data.

8A. Hybrid contextual and statistical method

8A.1 Issues

Identifying GNs and DANs is a challenging task due to the following problems: first, authors occasionally ignore the predefined formats resulting in many variations and inconsistencies (shortened, abbreviated, and slightly altered forms). Secondly, other terms that have a similar format such as protein names, non-PHS grant numbers, and even zip codes, often appear together.

Tables 8.1 and 8.2 show examples of GNs and DANs, and their variations. As shown in Table 8.1, GNs may be expressed in different ways even in a single sentence. In the first example, the first GN (1-P50-CA108786-01) has all components in the required format, but hyphens are inserted incorrectly between components. The second and third GNs (NS20023 and CA11898) include only two components: the administering organization and the serial number. The last GN (MO1 RR 30) has some missing components and the letter 'O' (in "MO1") is incorrectly used instead of zero. In examples 2 and 3, NIH and non-PHS (Department of Defense) grant numbers appear together. The last example shows a GN (CA97022) and a US zip code (CA 92037) having the same prefix and the same number of digits.

No	Example text
1	Supported by National Institutes of Health Grants No. **1-P50-CA108786-01**, **NS20023** and **CA11898** and by Grant No. **MO1 RR 30** through the General Clinical Research Centers Program, National Center for Research Resources, National Institutes of Health.
2	Supported by National Institutes of Health (NIH) grant no. **CA78657**, Department of Defense grant no. **BC010002**, Aging and Alzheimer Research Center grants (A.F.), and NIH grant no. **1CA76274** (R.B.).
3	This research was supported in part by DGAPA/UNAM (DirecciÃ³n General del Personal AcadÃ©mico/Universidad Nacional AutÃ³noma de MÃ©xico) **IN207503-3**, **IN206503-3** and **IX217404**, CONACyT (Consejo Nacional de Ciencia y TecnologÃ-a) **36505-N**, USDA (United States Department of Agriculture) **2002-35302-12539** and NIH (National Institutes of Health, U.S.A.) **1R01 AI066014-01**.
4	Supported by National Institutes of Health Grants **CA97022** and **GM68487**. To whom correspondence should be addressed: The Scripps Research Institute, Dept. of Immunology, SP231, 10550 N. Torrey Pines Road, La Jolla, **CA 92037**. Tel.: 858-784-7750; Fax: 858-784-7785; E-mail: klemke@scripps.edu.

Table 8.1. Examples of GNs (highlighted) and other technical terms (underlined)

Many variations in the DAN formats can also be found in the literature. The first example in Table 8.2 shows three sentences each of which has a *RefSeq* DAN (NT_011786, XM 658485, and NM177427). As mentioned in Section 4.3, *RefSeq* DAN must have a two-letter prefix followed by an underscore and six- or nine-digit number. However, we can see that the second and third *RefSeq* DANs have no underscore between the prefix and number. In addition, the presence of other technical or biological terms significantly increases the complexity of identifying DANs, as shown in the second example, in which *GenBank* DANs (highlighted) and other terms (underlined) appear together.

No	Example text
1	To identify new isoforms of AIF we first analyzed, by an in silico approach, human AIF (NCBI Gene Data Base accession number **NT_011786**, gene ID 9131). The nucleotide sequence of the pkcB mRNA were previously deposited as "Aspergillus nidulans FGSC A4 hypothetical protein" (AN5973.2; REFSEQ accession number. **XM 658485**). We named this newly discovered variant as P2X7-j because previous studies identified splice variants isoforms designated P2X7-b-P2X7-h (Ref. 25, accession numbers AY847 (298-304)), and a truncated P2X7 variant 2 (149 residues) (Ref. 26, accession number **NM177427**).
2	Sequences from this study have been deposited in GenBank under accession numbers **CY003847** to **CY006042**. This work was supported by the American Lebanese Syrian Associated Charities, a Cancer Center Support Grant (**CA 21765**), the U.S. Public Health Service (grant **AI95357**), and the Hartwell Foundation. Identical sequences were found for five strains: **MDA2833**, **MDA0990**, **HUMC1166**, **CCUG38963**, and **CCUG50611**. Queries through GenBank BLAST showed that the organisms most closely matched N. meningitidis (GenBank accession no. **AL162758** and many others) at 95.7% (1,410 of 1,473 bp).

Table 8.2. Examples of DANs (highlighted) and other technical terms (underlined)

Moreover, new types of GNs and DANs are added periodically; for instance in 2006 and 2007, *ISRCTN* and *PubChem* were newly included in MEDLINE. These new types may have significantly different formats such as a newly created organizational code in GNs, new prefixes in DANs, and/or different numbers of digits in the serial number.

8A.2 Previous work

Rule-based methods based on heuristics and domain-specific word/pattern dictionaries are the most conventional approach for term identification [54, 55]. However, hand-crafted rules cannot readily deal with the many variations and inconsistencies in GNs and DANs due to the lack of generalization capability. To overcome the limitation of such rule-based methods, statistical approaches based on word distributions have been developed [56, 57]. However, these statistical methods often yield unreliable results in the analysis of biomedical text if large and representative training datasets are not available, the number of words included in a given test sentence or abstract is inadequate, or certain technical words closely related to a specific term appear infrequently. At a later stage we intend to investigate machine learning techniques such as SVM, Hidden Markov Model (HMM), etc. that have been reported to show a good performance in biomedical named entity recognition, although the time-consuming task of building a large annotated training corpus would be essential for this [58, 59].

8A.3 Our approach

We present a hybrid approach based on contextual and statistical information to automatically identify GNs and DANs. Our method first broadly extracts potential candidates for GNs and DANs using a rule-based method, and then calculates the confidence score for each candidate based on the relative frequency of occurrence of all individual words found in the sentence in which the candidate term appears (called “GN sentence” or “DAN sentence”). Such statistical information is based on the sentence-level word frequency, i.e., the number of times a given word appears in a GN or DAN sentence, and is estimated using a training dataset obtained from biomedical articles in NLM’s PubMed Central as well as those downloaded from publisher Web sites. This confidence score is positively or negatively weighted depending on morphological cues and contextual information (explained later), to offset statistical errors.

Figure 8.1 illustrates an overview of our method. First, the HTML-formatted body text of an article is segmented into text zones, and the zones that contain clues indicating the existence of GNs and DANs are located and labeled as “GN zone” and “DAN zone” by the zoning and labeling modules described in Sections 6 and 7. Next, we extract candidates for GNs and DANs by applying the predefined rules. For each candidate, the corresponding confidence score is calculated using contextual and statistical information. The candidates with confidence scores exceeding a predefined threshold are submitted to a human operator for final verification.

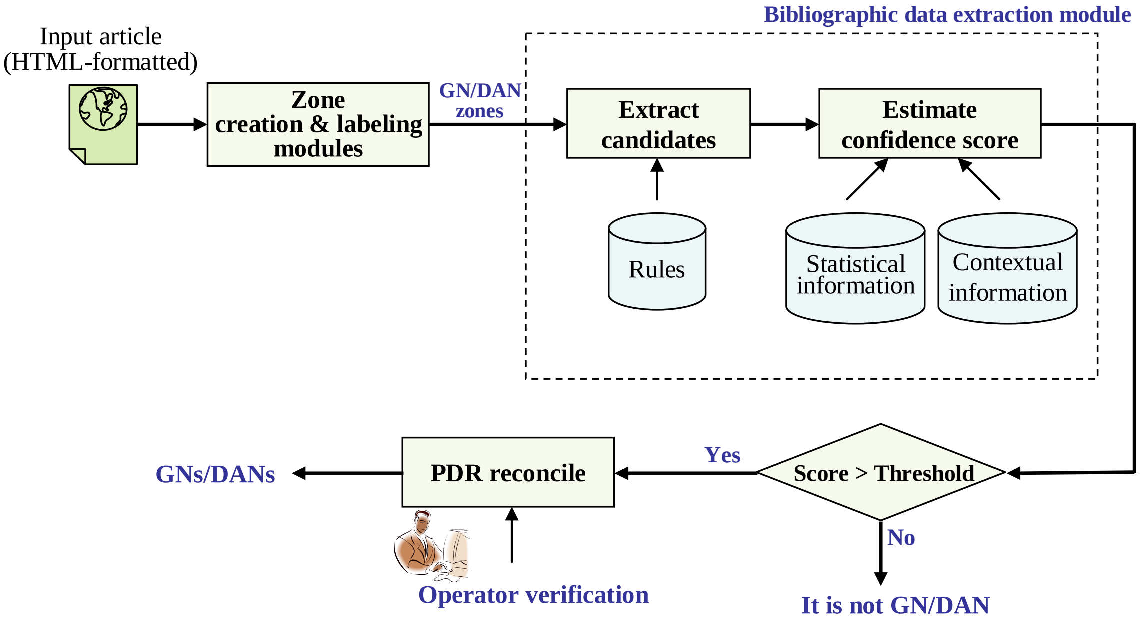


Figure 8.1. An overview of the automated system to extract GNs and DANs

8A.3.1 Rules for extracting candidates for GNs and DANs

Potential candidates for GNs are extracted from the GN zones using the following rules reflecting the aforementioned characteristics of GNs and their variations. A candidate word or word string must:

1) consist of capital letter(s) and two or more consecutive numerals.

2) consist of five or more consecutive numerals with/without lowercase letters.

3) accept SPACE, '-', and '/' as legitimate components.
4) be at least four characters long.
5) have three or more numerals.

Rules 4 and 5 are applied to the candidates that satisfy rule 1.

Similar rules apply to the extraction of candidates for DANs.

8A.3.2 Estimation of confidence score

Once all potential candidates for GNs and DANs are extracted, their confidence scores are calculated based on statistical and contextual information. Statistical information consists of the sentence-level frequency of the words in sentences containing GNs or DANs. To estimate this word frequency reliably, we first created a large volume of training data consisting of 12,760 online biomedical journal articles that were indexed by MEDLINE in 2006 and found to have GNs/DANs in their body text (11,107 articles for GNs and 1,653 articles for DANs). Next, we extract the text zones containing GNs/DANs from the body text of each article in the training dataset. Usually, such text zones consist of several sentences of which at least one contains GNs or DANs. Finally, we build a dictionary consisting of words appearing in GN/DAN sentences, and estimate their frequency of occurrence in the GN/DAN sentences and "other sentences".

Now we derive an expression for the confidence score for GN or DAN candidates. Let $N_n(c_g, w_x)$ and $N_n(c_o, w_x)$ be the number of occurrences of word w_x in a GN/DAN sentence class c_g and "other sentence" class c_o, respectively. The probability that w_x has occurred in class c_g is then estimated by the conditional relative frequency:

$$P(w_x \mid c_g) = \frac{N_n(c_g, w_x)}{N_n(c_g)} \qquad (1)$$

where, $N_n(c_g) = \sum_i N_n(c_g, w_i)$. Similarly, $P(w_x \mid c_o)$, the conditional probability that w_x has occurred in class c_o is also estimated. The conditional probabilities for a sentence S_x consisting of a set of words, $\{w_1, w_2, \ldots w_k\}$ in class c_g and c_o are equal to the product of the conditional probabilities of individual words by making the Naïve Bayesian assumption that all words in the sentence are independent of each other.

$$\begin{aligned} P(S_x \mid c_g) &= P(w_1, w_2, \ldots w_k \mid c_g) \\ &= \prod_{x=1}^{k} P(w_x \mid c_g) \end{aligned} \qquad (2)$$

Assuming that the words closely related to GN/DAN have a high relative frequency score in the GN/DAN sentence class, we estimate the confidence score of a given GN/DAN candidate, CAN_x found in the sentence S_x based on the difference between $P(S_x \mid c_g)$ and $P(S_x \mid c_o)$. Finally, the confidence score of the candidate, CAN_x normalized into the range from 0 to 1 is obtained by applying the sigmoid normalization described in [60].

$$Conf(CAN_x) = \frac{1}{1 + \exp\{-\alpha(l_x(c_g) - l_x(c_o))\}} \qquad (3)$$

where $l_x(c_g) = \log P(S_x \mid c_g)$ and $l_x(c_o) = \log P(S_x \mid c_o)$ are employed to avoid a floating-point underflow. This underflow results from the fact that conditional probabilities of all individual words in sentence S_x are less than 1. The factor α is the steepness parameter in the sigmoid function whose value is empirically determined.

Such a confidence score based on word frequency alone can often be unreliable when the number of words in the sentence is inadequate, or when certain technical words closely related to GNs and DANs occur infrequently, thereby resulting in false rejection errors. To avoid this problem, the confidence score is positively or negatively weighted depending on morphological and contextual information embedded in GN and DAN zones. Morphological cues are the standard format and prefix of GNs and DANs mentioned in the previous section. Contextual information depends on specific keywords and phrases strongly suggesting the existence of GNs and DANs in a sentence. Examples of these are: “National Institutes of Health”, “supported by”, “accession number”, and databank names, etc., which were collected by analyzing our training dataset. So a candidate would be positively weighted if it has the correct format and prefix, or its surrounding sentence has a word or phrase such as the ones listed above. The highest scoring candidates, the ones exceeding a set threshold (a particular score value) are presented to a human operator for final verification.

8A.3.3 Experimental results

We evaluate the performance of our proposed method in terms of recall rate. A test dataset consisting of 10,237 HTML-formatted online articles from over 52 different biomedical journal titles is created for evaluating our method. 8,982 articles from this set are used for extracting GNs, and the remaining 1,255 articles for the eleven types of DANs. Our experiments show that most GNs and DANs can be successfully identified, with recall rates of 99.8% and 99.6%, respectively, when the threshold is set to a confidence score of 5.

Despite this generally good performance, an analysis of the experimental results reveals certain types of rarely occurring errors. For example, in Table 8.3 (a), “290-02-0024” was not recognized as a GN correctly due to the following problems: a) the sentence containing this GN consists largely of words that are not closely related to GNs but commonly found in other parts of the body text of many biomedical documents, or as parts of a specific organization name (e.g., “Oregon Evidence-Based Practice Center”), thereby generating a low confidence score of 2, and b) at the time of this experiment, we had not included the name of the agency shown in our list of keywords and phrases (nor the format of GNs issued by this agency). The consequent lack of contextual and morphological information in this example results in lowering the confidence score. Conversely, other biomedical or technical terms such as “1057” in Table 8.3 (b) can be misrecognized with a high confidence score (of 10 in this case), when they follow the accepted formats, and when genuine GNs or DANs are also found within the same sentence.

While high recall rate is critical so as not to eliminate true GN/DAN candidates for operator review, the extraction of false candidates (low precision) reduces operator efficiency at the verification step. Future work is planned to employ a machine learning method such as SVM or HMM to reduce the false alarm errors, thereby improving overall performance and further reducing the human labor required for correction.

Sentence	This study was conducted by the Oregon Evidence-Based Practice Center under contract to the Agency for Healthcare Research and Quality (Rockville, MD) contract **290-02-0024**, Task Order 2.
Candidates (**Confidence score**)	220-02-0024 (**2**)

(a)

Sentence	We have previously calculated a quasi-atomic resolution model of the echovirus (EV) type 12Â·receptor complex based on cryo-negative stain transmission electron microscopy and image reconstruction of EV12 bound to a fragment of DAF comprising SCR3 and SCR4 (DAF34) (EM Data Bank code **1057** and Protein Data Bank code **1UPN** [PDB]) (21).
Candidates (**Confidence score**)	PDB/1057 (**10**), PDB/1UPN (**10**)

(b)

Table 8.3. Examples of (a) false rejection, and (b) false alarm errors

8B. Support Vector Machine to identify "Comment-on" (CON) data

8B.1 Issues

In this section, we describe research toward the automated extraction of CON data. Currently, NLM operators create this data manually, a process that requires an operator to first open an input article in a Web browser and read it for linguistic or contextual clues that suggest CON data. If these clues are found, the operator identifies the corresponding CON article from the reference section of the input article, and uses author names and title (of the CON article) in a PubMed search which returns the PubMed ID (PMID) of the CON article. This manual process is time-consuming, and success mainly depends on the operators' linguistic knowledge and understanding of scientific expressions and writing styles.

8B.2 Proposed approach

In order to minimize the manual effort and to improve accuracy and processing speed, we propose an automated identification method based on SVMs. Figure 8.2 shows an overview of the proposed method. Note that in a scientific article, all external sources (e.g., journal articles, books, or Web links) listed in the reference section are generally cited at least once within sentences ("citation sentences") in the body of the paper. From this observation, our method starts with detecting all "citation sentences" in the body text containing hyperlinks or specific tags such as square or round parentheses enclosing reference numbers. Next, the "CON sentences" that mention CON articles are identified from these "citation sentences" using the SVMs. The title and author names of the CON articles are then extracted from the reference section and sent to PubMed to acquire the corresponding PMIDs which are sent to the PDR Reconcile system for operator verification. The operator is also provided the "CON sentences" and the corresponding references to help with a final decision. The detailed procedure for identifying the CON list is described in the next section.

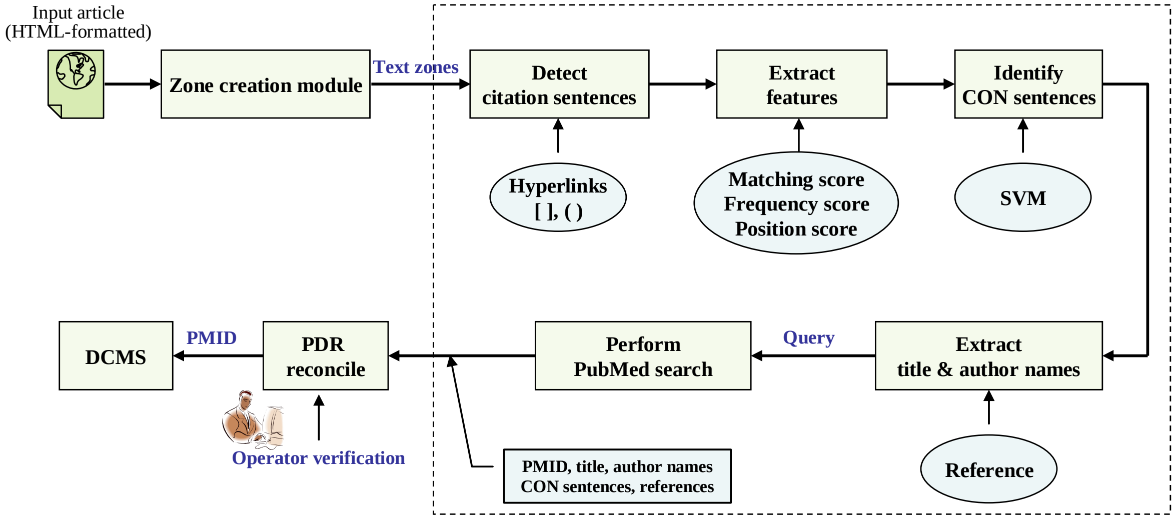


Figure 8.2. Overall procedure for automated identification of CON data.

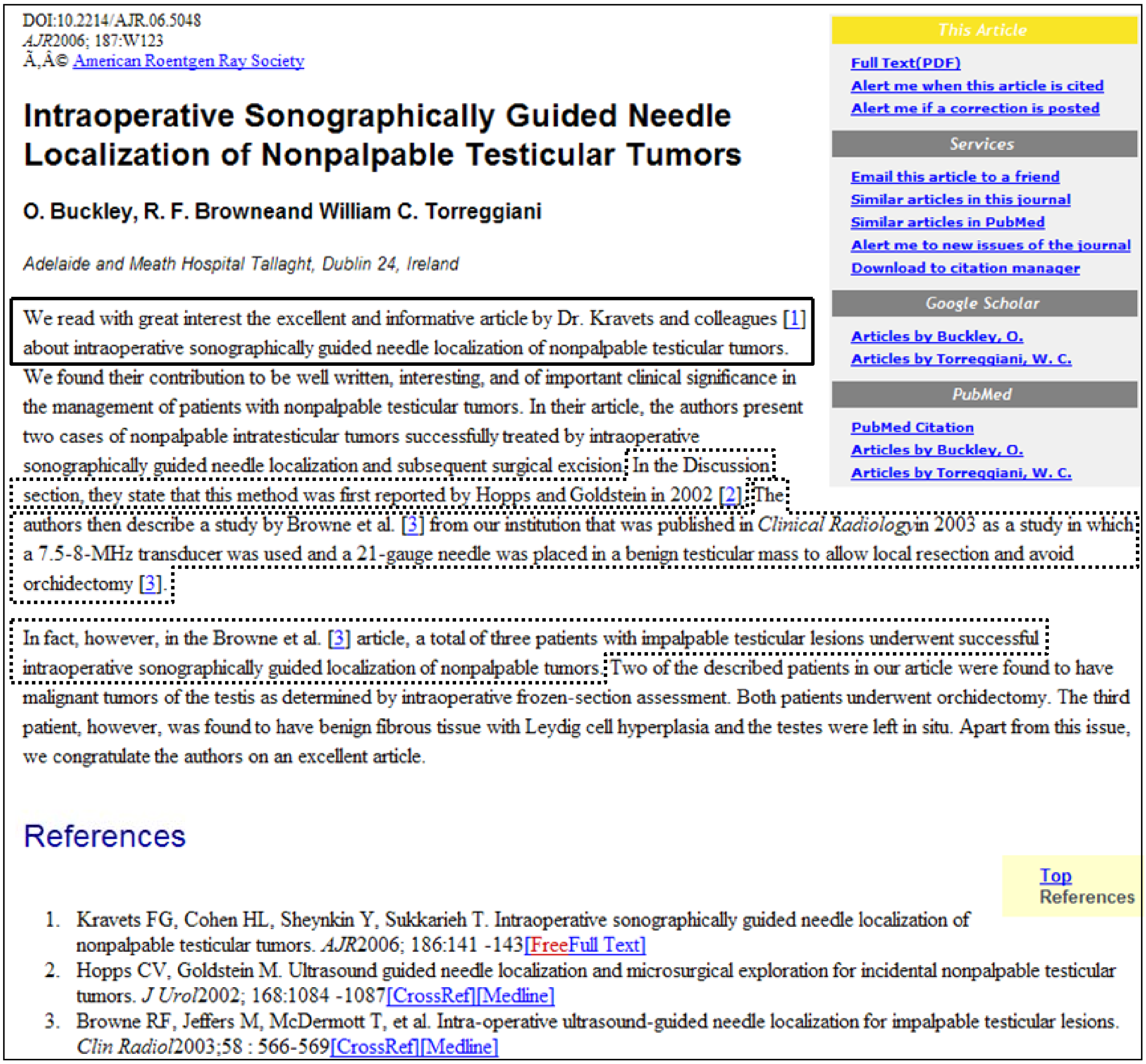
8B.3 Method

8B.3.1 Detection of "citation sentences"

As already noted, each "citation sentence" in a scientific article usually contains a tag (such as "(1)" or "[1]") which we may call an "in-text citation". These tags point to the bibliographical description of the cited source in the reference section. The sentence containing the "in-text citation" that specifically indicates the article commented on by the given article is defined as the "CON sentence". A "CON sentence" is therefore a subset of a "citation sentence". Figure 8.3 shows an example of an article with three "citation sentences" (dotted boxes) and a separate "CON sentence" (solid box). Our method first detects "citation sentences" from the body text, and then uses SVMs to identify "CON sentences" from these.

Very often, in HTML-formatted online articles, an "in-text citation" is hyperlinked to the corresponding external source as shown in Figure 8.4(a), the hyperlink consisting of both a source anchor (in the body text) and a destination anchor (in the reference section). The source anchor specified by "*A*" HTML element with "*href*" attribute appears at the "in-text citation" and points to the destination anchor. The destination anchor specified by "*A*" element with "*name*" attribute can be found at the beginning of the reference. The source anchor and its destination anchor have the same unique name, in this case, "REF1". By recognizing this anchor name, we can reliably detect the associated "citation sentence".

Unlike HTML articles, PDF documents (even when converted to HTML) do not have such hyperlink information. In this case, therefore, we detect "citation sentences" by recognizing the tags indicating "in-text citations" such as a pair of square or round brackets enclosing reference numbers (Figure 8.4(b)), or author names and publication year (Figure 8.4(c)), or by tracking a superscript HTML tag pair enclosing reference numbers (Figure 8.4(d)).


This Article
Full Text(PDF)
Alert me when this article is cited
Alert me if a correction is posted
Services
Email this article to a friend
Similar articles in this journal
Similar articles in PubMed
Alert me to new issues of the journal
Download to citation manager
Google Scholar
Articles by Buckley, O.
Articles by Torreggiani, W. C.
PubMed
PubMed Citation
Articles by Buckley, O.
Articles by Torreggiani, W. C.
Top
References

DOI:10.2214/AJR.06.5048
*AJR*2006; 187:W123
Ã‚Â© American Roentgen Ray Society

Intraoperative Sonographically Guided Needle Localization of Nonpalpable Testicular Tumors

O. Buckley, R. F. Browneand William C. Torreggiani

Adelaide and Meath Hospital Tallaght, Dublin 24, Ireland

We read with great interest the excellent and informative article by Dr. Kravets and colleagues [1] about intraoperative sonographically guided needle localization of nonpalpable testicular tumors. We found their contribution to be well written, interesting, and of important clinical significance in the management of patients with nonpalpable testicular tumors. In their article, the authors present two cases of nonpalpable intratesticular tumors successfully treated by intraoperative sonographically guided needle localization and subsequent surgical excision. In the Discussion section, they state that this method was first reported by Hopps and Goldstein in 2002 [2]. The authors then describe a study by Browne et al. [3] from our institution that was published in *Clinical Radiology*in 2003 as a study in which a 7.5-8-MHz transducer was used and a 21-gauge needle was placed in a benign testicular mass to allow local resection and avoid orchidectomy [3].

In fact, however, in the Browne et al. [3] article, a total of three patients with impalpable testicular lesions underwent successful intraoperative sonographically guided localization of nonpalpable tumors. Two of the described patients in our article were found to have malignant tumors of the testis as determined by intraoperative frozen-section assessment. Both patients underwent orchidectomy. The third patient, however, was found to have benign fibrous tissue with Leydig cell hyperplasia and the testes were left in situ. Apart from this issue, we congratulate the authors on an excellent article.

References

1. Kravets FG, Cohen HL, Sheynkin Y, Sukkarieh T. Intraoperative sonographically guided needle localization of nonpalpable testicular tumors. *AJR*2006; 186:141 -143[FreeFull Text]
2. Hopps CV, Goldstein M. Ultrasound guided needle localization and microsurgical exploration for incidental nonpalpable testicular tumors. *J Urol*2002; 168:1084 -1087[CrossRef][Medline]
3. Browne RF, Jeffers M, McDermott T, et al. Intra-operative ultrasound-guided needle localization for impalpable testicular lesions. *Clin Radiol*2003;58 : 566-569[CrossRef][Medline]

Figure 8.3. An online biomedical article showing "citation sentences" (dotted boxes) and a "CON sentence" (solid box).

Hyperlink (Source anchor)	We read with great interest the excellent and informative article<SUP> </SUP>by Dr. Kravets and colleagues [**<A HREF="#REF1">**1</A>] about intraoperative sonographically<SUP> </SUP>guided needle localization of nonpalpable testicular tumors.<SUP> </SUP>We found their contribution to be well written, interesting,<SUP> </SUP>and of important clinical significance in the management of patients<SUP> </SUP>with nonpalpable testicular tumors.
Hyperlink (Destination anchor)	<OL COMPACT> **<A NAME="REF1">**<!-- null --></A><LI VALUE=1> Kravets FG, Cohen HL, Sheynkin Y, Sukkarieh T. Intraoperative sonographically guided needle localization of nonpalpable testicular tumors. <I>AJR</I> 2006; 186:141 -143<!-- HIGHWIRE ID="187:1:W123:1" --><a href="/cgi/ijlink?linkType=FULL&journalCode=ajronline&resid=186/1/141" ><nobr>[<font COLOR="CC0000">Free</font>&nbsp;Full&nbsp;Text]</nobr></a>

(a)

Text symbols of in-text citation	The combination of improved bone marrow transplantation techniques, now called HSC transplantation, supportive animal data **[2]** and coincidental observations (improvement in coexisting autoimmune disease after HSC transplantation for conventional indications, such as aplastic anaemia, leukaemia and cancer **[3]**) has allowed the concept to move forward to the clinic.

(b)

Author names & publication year	For collections of small families, computer programs such as GeneHunter **(Kruglyak et al. 1996)** and Merlin **(Abecasis et al. 2002)** calculate the correct variance of the sharing statistic, conditional on all observed marker information; therefore, the correct test statistic and P value are computed

(c)

HTML tags (superscript)	<SPAN CLASS="ps1p14"><NOBR><SPAN CLASS="ft3">Evidence continues to accumulate that children in families where there</SPAN></NOBR></SPAN><SPAN CLASS="ps1p15"><NOBR> <SPAN CLASS= "ft3"> is sub- stance &#160;abuse, &#160; including &#160;alcohol, &#160;have &#160;a &#160;number of &#160; difficulties. <SPAN CLASS= "em1p5">**<SUP>1</SUP>**</SPAN> &#160;The rate &#160;</SPAN></NOBR></SPAN><SPAN CLASS="ps1p16"><NOBR><SPAN CLASS="ft3">of externalizing and internalizing problem in children of substance abusers appears </SPAN></NOBR></SPAN><SPAN CLASS= "ps1p17"><SPAN CLASS="ft3">to be high.<SPAN CLASS="em1p5">**<SUP>3</SUP>**</SPAN>

(d)

Figure 8.4. Detection of "citation sentences" using (a) hyperlink information, (b) text symbols of "in-text citation", (c) author names and publication year, and (d) superscript HTML tags.

8B.3.2 Feature extraction

Once "citation sentences" are extracted, feature vectors are calculated for training and testing SVMs. We define a 30-dimension binary feature vector created by combining three types of features which have been experimentally found to be effective to represent a "CON sentence". The first feature is a score that quantifies the degree to which a cue phrase (examples shown in Table 8.4) matches the words in a "CON sentence". We have collected 52 cue phrases from an analysis of hundreds of samples of "CON sentences". The matching score for a sentence s_i is defined as follows:

$$M(s_i) = \max \frac{w_c(s_i)}{W_c + w_g(s_i)} \qquad c = 1, 2, 3, ..., 52 \qquad (1)$$

Here, W_c and $w_c(s_i)$ represent the total number of words contained in a cue phrase *c* and the number of words in this phrase that are actually found in s_i, respectively. $w_g(s_i)$ is the number of words that remain unmatched. Assume that a sentence starts with "*I read with **great** interest …*" and we have "*I read with interest*" in our list of predefined cue phrases. This cue phrase consists of four words (W_c = 4) which are all found in s_i ($w_c(s_i)$ = 4). However, this phrase does not strictly match the word string in s_i which has the additional word "*great*" ($w_g(s_i)$=1). In this case, the resulting matching score $M(s_i)$ is $\frac{4}{4+1} = 0.8$.

Cue phrases	The article (paper, letter, study, research) by … I (We) read with interest … In the editorial … would like to reply (comment) to … In this issue … In their recent (article, letter, paper, report) … Reply (respond) to the comment (s)

Table 8.4. Samples of cue phrases

The second feature is based on sentence position, since typically "CON sentences" are located at the beginning of the body text. Such position information can serve as a good feature to distinguish a "CON sentence" from other "citation sentences". Position of a sentence is expressed as

$$P(s_i) = 1 - \frac{D_n(s_i)}{D_N} \qquad (2)$$

where D_N is the total number of characters in the given document *D*, and $D_n(s_i)$ is the number of characters located before the sentence s_i.

The third and last feature is the frequency of occurrence of the names of authors (term frequency) of CON articles, based on our observation that authors of articles commented on are more frequently mentioned in the text. The frequency score of author names of external sources listed in the reference section is defined as follows:

$$TF(a_i) = \frac{tf(a_i, D)}{tf_{\max}(a, D)} \qquad (3)$$

where $tf(a_i, D)$ and $tf_{\max}(a, D)$ denote the number of occurrences of author a_i and the maximum number of occurrences of author names in the given document *D*, respectively.

Each of these three features is quantized and normalized to a real value ranging between 0 and 1. The normalized real-valued features are converted to a 10-bit binary vector (*i*-th bit position corresponds to real values between *i*/10 and (*i*+1)/10). For example, 0.35 is converted to "0001000000". These three 10-bit binary vectors are then concatenated to produce the 30-dimensional feature vector to be used as input to the SVM to detect "CON sentences".

8B.3.3 SVM to identify "CON sentences"

We implemented two types of SVMs: one with a polynomial kernel function and the other with a RBF kernel function using the MYSVM, a free software package for non-commercial use developed by Rüping (at University of Dortmund) [61]. These two kernel functions defined in equations (4) and (5) below have been most commonly used in SVM-based pattern recognition applications. We evaluate their recognition performance using real online biomedical journal articles.

$$K(x, y) = (x \cdot y + 1)^p \tag{4}$$

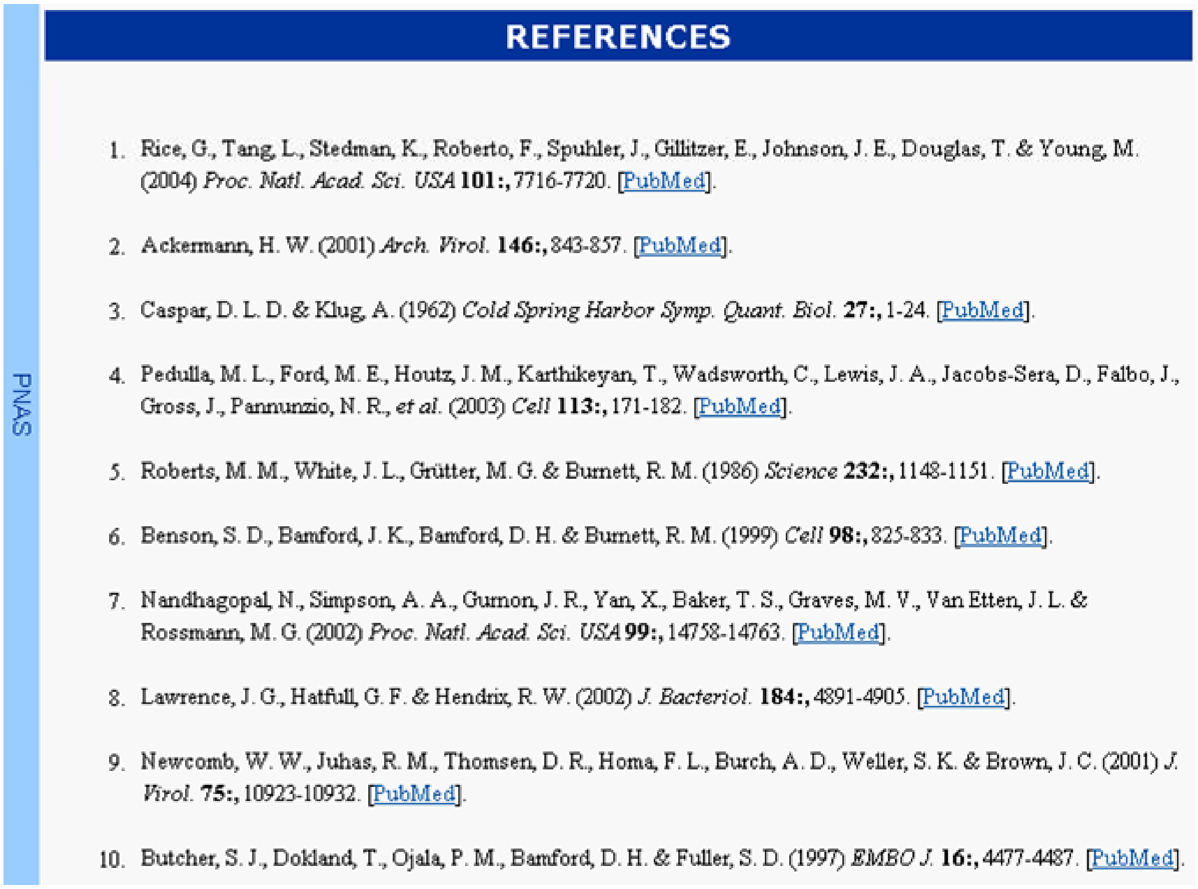
$$K(x, y) = \exp(-\gamma \|x - y\|^2) \tag{5}$$

8B.3.4 "Significance" measure: an alternative approach

As a slight detour, we introduce an alternative method that recognizes "CON sentences" by estimating the "significance of sentence" using a predefined score function, and compare this method to the SVM approach. Measuring the significance of a given sentence using a score function is commonly done in automatic text summarization [62, 63]. Here, we define and implement three types of score functions based on the three basic features mentioned previously, namely: the position of sentence, author name statistics, and linguistic and contextual information. We also integrate the three individual score functions as a fourth approach. At the learning stage, the threshold value of each score function is calculated using the training dataset (the same as that used for SVM training). As a result, each score function shows the best performance in identifying "CON sentence" for the training dataset using that particular threshold. At the recognition stage, any "citation sentence" that has a significance value larger than the threshold calculated at the learning stage is labeled as a "CON sentence".

8B.3.5 PMID acquisition through PubMed search

Once the "CON sentence" is identified by SVM, the title and authors of the CON article can be extracted from the reference section (we have a list of pairs of "citation sentences" and their corresponding external sources in reference section, obtained in the preceding step of extracting "citation sentences"). The title and authors are then used in a PubMed search to retrieve the PMID of the CON article. However, articles in certain journals such as *Proceedings of the National Academy of Sciences (PNAS)* do not provide titles of external sources in their reference sections as shown in Fig. 8.5. In such cases, we extract pagination and publication date instead of article title. In order to access PubMed for sending a query consisting of author names and title (or pagination and publication date), and retrieving XML-formatted search results, we employ the *EUtilities* provided by NCBI. The PMID for an identified CON article is finally extracted by locating a tag pair of "<CommentOn>" and "</CommentOn>" in the PubMed search result.



REFERENCES

1. Rice, G., Tang, L., Stedman, K., Roberto, F., Spuhler, J., Gillitzer, E., Johnson, J. E., Douglas, T. & Young, M. (2004) *Proc. Natl. Acad. Sci. USA* **101:,** 7716-7720. [PubMed].
2. Ackermann, H. W. (2001) *Arch. Virol.* **146:,** 843-857. [PubMed].
3. Caspar, D. L. D. & Klug, A. (1962) *Cold Spring Harbor Symp. Quant. Biol.* **27:,** 1-24. [PubMed].
4. Pedulla, M. L., Ford, M. E., Houtz, J. M., Karthikeyan, T., Wadsworth, C., Lewis, J. A., Jacobs-Sera, D., Falbo, J., Gross, J., Pannunzio, N. R., *et al.* (2003) *Cell* **113:,** 171-182. [PubMed].
5. Roberts, M. M., White, J. L., Grütter, M. G. & Burnett, R. M. (1986) *Science* **232:,** 1148-1151. [PubMed].
6. Benson, S. D., Bamford, J. K., Bamford, D. H. & Burnett, R. M. (1999) *Cell* **98:,** 825-833. [PubMed].
7. Nandhagopal, N., Simpson, A. A., Gurnon, J. R., Yan, X., Baker, T. S., Graves, M. V., Van Etten, J. L. & Rossmann, M. G. (2002) *Proc. Natl. Acad. Sci. USA* **99:,** 14758-14763. [PubMed].
8. Lawrence, J. G., Hatfull, G. F. & Hendrix, R. W. (2002) *J. Bacteriol.* **184:,** 4891-4905. [PubMed].
9. Newcomb, W. W., Juhas, R. M., Thomsen, D. R., Homa, F. L., Burch, A. D., Weller, S. K. & Brown, J. C. (2001) *J. Virol.* **75:,** 10923-10932. [PubMed].
10. Butcher, S. J., Dokland, T., Ojala, P. M., Bamford, D. H. & Fuller, S. D. (1997) *EMBO J.* **16:,** 4477-4487. [PubMed].

Fig. 8.5. Reference section of an article in PNAS

8B.4 Experimental results

In this section, we provide evaluation results of our approaches to identify "CON sentences". However, the specific implementation to obtain PMID of the identified CON article is currently under study and will be reported at a later time.

To build a dataset for the recognition experiments, we collected 1,236 "citation sentences" from 175 HTML-formatted online articles. These articles appear in 24 different biomedical journals, and their publication types are Letter (62.3%), Review (5.1%), Editorial (4.0%), and Commentary (28.6%). Of the 1,236 "citation sentences", we randomly select 641 to train the SVMs, and to calculate the threshold for the score functions. The remaining 595 are used as the test set in the experiments.

We evaluated the performance of SVMs and score functions (“significance of sentence” measure) in terms of precision, recall, and F-measure rates.

Our experiments show that the SVM with polynomial kernel function yields the best performance in terms of recall (97.06%) and F-measure (97.06%) rates, as shown (bolded) in Table 8.5. The SVM with RBF kernel function shows lower recall and F-measure rates than the score functions, though it yields a slightly better precision rate than the SVM with polynomial function. Among the score function-based methods, the integrated method shows the best performance overall, as expected. Therefore we conclude that the SVM with polynomial kernel function is the most appropriate scheme for identifying “CON sentences”.

	Precision (%)	Recall (%)	F-Measure (%)
Score function 1 (sentence position)	65.35	64.71	65.02
Score function 2 (author name statistics)	83.19	97.06	89.59
Score function 3 (linguistic information)	85.71	52.94	65.45
Score function 4 (integrated method)	91.59	96.08	93.78
SVM with RBF	97.80	87.25	92.22
SVM with polynomial	**97.06**	**97.06**	**97.06**

Table 8.5. Precision, recall, and F-Measure rates of SVMs and score functions

9. Next steps

In this section we present some ongoing and future activities.

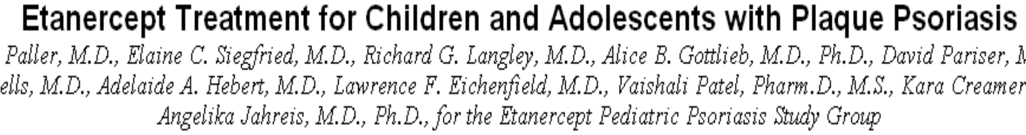
Combine labeling techniques. Two automated labeling techniques, a Naïve Bayesian algorithm with if-then-else rules and a Support Vector Machine, have been presented in Section 7. Both have been found to perform well, but each has strengths and weaknesses. The goal in this effort is to combine them to exploit the advantages of each.

As discussed in Section 7, the Naïve Bayesian algorithm is robust with respect to typographic and other errors since it is based on statistics and keywords, but has a problem dealing with rare or exceptional cases. This problem can be addressed by the rule-based algorithm used to supplement the Naïve Bayesian algorithm, but rules have to be built manually, and at any given time may not be adequate. The second technique, the Support Vector Machine, is robust with respect to the "curse of dimensionality" as well as to correlated features, but it is biased toward the class with more training samples. Since each technique has advantages and disadvantages, we plan to combine these techniques, compute confidence values normalized across techniques, and select the output (zone label) from the technique that gives the higher confidence.

Collect training data automatically. The success of supervised learning methods relies on the availability of adequate training samples. This has been a problem in the case of certain bibliographic data, such as grant support information (GS). GS is currently detected by a Naïve Bayesian algorithm with if-then-else rules. The algorithm relies on limited features: zones labeled as containing grant numbers, and the occurrence of "support words." In contrast to other bibliographic data (titles, authors, DANs, GNs), there are no explicit clues in the GS field in existing MEDLINE citations that suggest how GSs were selected. As a result it is difficult to create suitable training data to design a reliable supervised learning algorithm for GS detection.

We therefore plan to use *unsupervised* learning methods to collect training samples. To select a suitable method, we will possibly investigate the K-means clustering algorithm, the self-organizing map (SOM) and the expectation-maximization algorithm. Our approach will take unlabeled zones in articles from different publishers, and use the selected unsupervised learning method to group these into clusters. These clusters will be classified as either GS or not based on (1) the analysis of clusters' centroids in the feature space or (2) the majority of members in the clusters labeled as GS using the current Naïve Bayesian algorithm with if-then-else rules. The GS clusters will then provide the samples to train the supervised learning methods we currently use for better detection of grant support information.

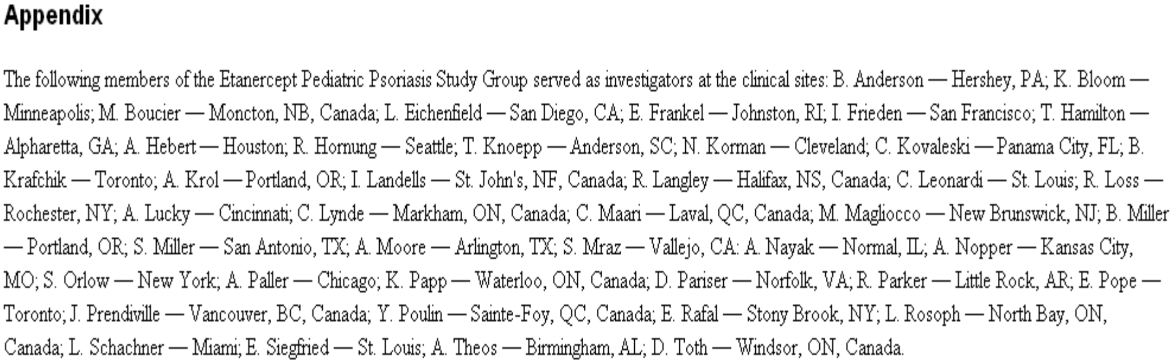
Extract Investigator Names listed in the article. A new field has recently been defined in MEDLINE: Investigator Name. This is a reflection of the increasingly collaborative nature of biomedical research involving many people, possibly from different organizations. While such people are not "authors" of the article, they are listed as investigators, particularly when the article has a "corporate author," usually an organization or study group. An example of this is from the January 17, 2008 issue of New England Journal of Medicine:



Etanercept Treatment for Children and Adolescents with Plaque Psoriasis

Amy S. Paller, M.D., Elaine C. Siegfried, M.D., Richard G. Langley, M.D., Alice B. Gottlieb, M.D., Ph.D., David Pariser, M.D., Ian Landells, M.D., Adelaide A. Hebert, M.D., Lawrence F. Eichenfield, M.D., Vaishali Patel, Pharm.D., M.S., Kara Creamer, M.S., Angelika Jahreis, M.D., Ph.D., for the Etanercept Pediatric Psoriasis Study Group

The corporate author is the last item above: *Etanercept Pediatric Psoriasis Study Group*. The text at the end of this article states: "The investigators in the Etanercept Pediatric Psoriasis Study Group are listed in the Appendix." The Appendix containing the investigator names appears in the article as:



Appendix

The following members of the Etanercept Pediatric Psoriasis Study Group served as investigators at the clinical sites: B. Anderson — Hershey, PA; K. Bloom — Minneapolis; M. Boucier — Moncton, NB, Canada; L. Eichenfield — San Diego, CA; E. Frankel — Johnston, RI; I. Frieden — San Francisco; T. Hamilton — Alpharetta, GA; A. Hebert — Houston; R. Hornung — Seattle; T. Knoepp — Anderson, SC; N. Korman — Cleveland; C. Kovaleski — Panama City, FL; B. Krafchik — Toronto; A. Krol — Portland, OR; I. Landells — St. John's, NF, Canada; R. Langley — Halifax, NS, Canada; C. Leonardi — St. Louis; R. Loss — Rochester, NY; A. Lucky — Cincinnati; C. Lynde — Markham, ON, Canada; C. Maari — Laval, QC, Canada; M. Magliocco — New Brunswick, NJ; B. Miller — Portland, OR; S. Miller — San Antonio, TX; A. Moore — Arlington, TX; S. Mraz — Vallejo, CA: A. Nayak — Normal, IL; A. Nopper — Kansas City, MO; S. Orlow — New York; A. Paller — Chicago; K. Papp — Waterloo, ON, Canada; D. Pariser — Norfolk, VA; R. Parker — Little Rock, AR; E. Pope — Toronto; J. Prendiville — Vancouver, BC, Canada; Y. Poulin — Sainte-Foy, QC, Canada; E. Rafal — Stony Brook, NY; L. Rosoph — North Bay, ON, Canada; L. Schachner — Miami; E. Siegfried — St. Louis; A. Theos — Birmingham, AL; D. Toth — Windsor, ON, Canada.

It is estimated that in 2008 there will be about 5,000 articles with investigator names, with an average of 35 names per article, clearly a burden if manually entered. We are collecting samples of online articles that have corporate authors and investigators listed. Based on an analysis of these samples, we will design algorithms to identify and extract investigator names. A possible approach is to use the detected corporate author name and other clues to locate text zones containing investigator names, and then to apply a parsing algorithm to these zones using punctuation-based templates to extract the names. These templates can be predefined by analyzing the punctuation format patterns in samples of investigator names, or they can be dynamically created based on the current content of investigator names.

Locating and Parsing References in Online Articles. Even though a MEDLINE citation does not include references, these can nevertheless provide valuable hints to improve zone detection and also to extract key information for MEDLINE citations, such as CON data. References can also assist in assigning MeSH terms to an article through an analysis of MeSH terms already assigned to referenced articles. Detecting and analyzing references, therefore, is seen as a useful preprocessing step. We plan to develop statistical machine learning algorithms for locating and parsing references from HTML medical journal articles.

To locate references, we use the following features: (1) They contain distinctive text, e.g., author names, abbreviated journal names, pagination, publication years, etc.; (2) They have similar geometric features, e.g., occurring near the bottom of the page, having similar width and height, etc.; (3) All references are consecutive neighbors, and there must be a line-break between adjacent references. We therefore formulate reference detection as a two-class classification. After rendering the HTML article in a Browser and segmenting the pages, geometric and text features are extracted from the zones, and an SVM classifier is used to classify the zones as either reference or non-reference. The third observation above is a useful constraint which can expedite the process and increase its reliability.

For parsing references to obtain entities, such as author names, journal title, volume, pagination and publication year, we adopt a two-step process. The first step is a multiclass classification to assign each word in the reference an entity label. This classification requires local features of every word. In order to utilize the important correlation between adjacent words, these local features include the attributes of not only the word itself but also of its immediate neighbors. In addition, several rules may be formulated, despite many styles and variations found in references. Examples of such rules are: (1) The journal title must appear before volume and pagination (if these exist); (2) “J”, “J.”, or “Journal” cannot be labeled as an isolated single journal title entity; it must be accompanied by at least one of its adjacent neighbors to be part of the journal title. These rules are useful as global constraints, with which the label sequence must comply. In the second step in our algorithm, labels with low confidence are systematically corrected, if the entire label sequence violates the global rules.

10. Summary and conclusions

This report documents our development of a system, *Publisher Data Review* or PDR, that extracts bibliographic data (required MEDLINE citation fields) missing from the XML citation files that publishers routinely deliver to NLM. The technique relies on the extraction of these missing entities from online articles found on publishers' Web sites. The processing of these articles begins with segmenting each HTML page into zones, assigning a label to every zone containing data of interest, and then extracting the specific data for operator review and verification. The entities of current interest are: databank accession numbers (**DAN**), grant numbers (**GN**), grant support or category of funding institution (**GS**), and other articles commented on by the author ("commented-on" or **CON**).

Experimental results from evaluations of all our algorithms to date are summarized here.

Zoning. Our zoning algorithm based on geometric layout analysis by the X-Y cut technique correctly identified 9,376 zones out of 9,726 (from 104 articles appearing in 11 journal issues), for an accuracy figure of 96.4% (Section 6.)

Labeling. Our Naïve Bayesian algorithms were largely successful in labeling zones containing grant numbers (F-Measure = 98.78) and databank accession numbers (F-Measure = 96.87). While these results are good, we have devised heuristic if-then-else rules that will be used to supplement the Naïve Bayesian algorithms to deliver even higher performance.

We developed a second method for labeling, this one based on the Support Vector Machine. Out of 877 test zones containing DANs, this method labeled 821 of these correctly (accuracy = 93.6%). Reflecting the more consistent text in zones that contain grant numbers, of 1,224 such test zones containing GNs, 1,220 were labeled correctly (accuracy = 99.7%). Detailed evaluation results for both labeling methods are given in Section 7.

Entity extraction. As described in Section 8, the technique to extract GNs and DANs from the labeled zones is based on a hybrid contextual and statistical method, while a Support Vector Machine is used to identify CON data. The experimental results for GNs and DANs on a test dataset consisting of 10,237 online articles from over 52 different biomedical journal titles show that GNs and DANs are extracted with recall rates of 99.8% and 99.6%, respectively. For CON data, the evaluation conducted on 1,236 "citation sentences" from 175 online articles shows that our algorithm based on an SVM with polynomial kernel function can correctly identify CON data with an F-measure of 97.06 (same number for both precision and recall rate as well.)

As shown above, the evaluation of our algorithms indicate reasonably good performance, enabling us to complete an initial version of PDR, now in field testing at the NLM's Indexing Section. Early feedback from the testers is positive, but we believe additional improvements are possible by pursuing the tasks outlined in Section 9.

11. References

1. http://www.ncbi.nlm.nih.gov/Genbank/
2. http://clinicaltrials.gov/
3. http://www.ncbi.nlm.nih.gov/geo/
4. http://isrctn.org/
5. http://www.ncbi.nlm.nih.gov/RefSeq/
6. http://www.ncbi.nlm.nih.gov/Omim/
7. http://www.pdb.org/
8. http://pubchem.ncbi.nlm.nih.gov/
9. "Technical Memorandum 347: Databank Accession Numbers," National Institutes of Health, National Library of Medicine, October, 1993.
10. "Activity Codes, Organizational Codes, and Definitions used in Extramural Programs". (Available at http://grants.nih.gov/grants/funding/ac.pdf), National Institutes of Health, July, 2007.
11. "Technical Memorandum 447: Research Funding Support in DCMS and MEDLINE," National Institutes of Health, National Library of Medicine, October, 2006.
12. Y. Diao, H. Lu, S. Chen and Z. Tian, "Toward Learning Based Web Query Processing," Proc. of International Conference on Very Large Databases, 317-328 (2000).
13. S.-H. Lin, and J.-M. Ho, "Discovering Informative Content Blocks from Web Documents," Proc. of ACM SIGKDD, 588-593, (2002).
14. O. Buyukkokten, H. Garcia-Molina and A. Paepche, "Accordion Summarization for End-Game Browsing on PDAs and Cellular Phones," Proc. of the SIGCHI Conference on Human Factors in Computing Systems, 213-220, (2001).
15. E. Kaasinen, M. Aaltonen, J. Kolari, S. Melakoski and T. Laakko, "Two Approaches to Bringing Internet Services to WAP Devices," Proc. 9th International World Wide Web Conference, 231-246 (2000).
16. D. Cai, S. Yu, J.-R. Wen and W.-Y. Ma, VIPS: a Vision-Based Page Segmentation Algorithm, Microsoft Technical Report (MSR-TR-2003-79), 2003.
17. Nagy, G., Seth, S., and Viswanathan, M., A Prototype Document Image Analysis System for Technical Journals, Computer, vol. 25, pp. 10-22, 1992.
18. Ha, J., Haralick, R., and Phillips, I., Recursive X-Y Cut Using Bounding Boxes of Connected Components, Proc. 3rd International Conference Document Analysis and Recognition, pp. 952-955, 1995.
19. Baird, H.S., Jones, S.E., and Fortune, S.J., Image Segmentation by Shape-Directed Covers, Proc. International Conference Pattern Recognition, pp. 820-825, 1990.
20. O'Gorman, L., The Document Spectrum for Page Layout Analysis, IEEE Trans. Pattern Recognition and Machine Intelligence, vol. 15, pp. 1162-1173, 1993.
21. Jain, A.K. and Yu B., Document Representation and Its Application to Page Decomposition, IEEE Trans. Pattern Recognition and Machine Intelligence, vol. 20, no. 3, pp. 294-308, 1998.
22. Pavlidis, T., and Zhou, J., Page Segmentation and Classification, Graphical Models and Image Processing, vol. 54, pp. 484-496, 1992.
23. Hauser, S.E., Le D.X., and Thoma G.R., Automated zone correction in bitmapped document images, Proc. SPIE: Document Recognition and Retrieval VII, SPIE Vol. 3976, San Jose, CA, pp. 248-258, 2000.
24. Nagy, G., Twenty Years of Document Image Analysis in PAMI, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, no. 1, pp.38 – 62, 2000.
25. J. Zou, D. Le, G.R. Thoma, "Combining DOM tree and Geometric Layout Analysis for Online Medical Journal Article Segmentation," Proc. Joint Conference on Digital Libraries, 119-128 (2006).
26. J. Zou, D. Le, G.R. Thoma, "Online Medical Journal Article Layout Analysis," Proc. SPIE-IS&T Electronic Imaging 2007, SPIE Vol. 6500: 65000V (1-12), 2007.
27. J. Zou, D. Le, G.R. Thoma, "Structure and Content Analysis for HTML Medical Articles: A Hidden Markov Model Approach," Proc. ACM DocEng, pp. 199-201, 2007.
28. J. Kim, D. Le, and G. Thoma, **"**Automated labeling of bibliographic data extracted from biomedical online journals," *Proc. SPIE Electronic Imaging*, Vol. 5010, January, 2003, pp. 47-56.
29. J. Kim, D. Le, and G. Thoma, "Automated Labeling of Biomedical Online Journal Articles**,"** *Proc. 9th World Multiconference on Systemics, Cybernetics and Informatics*, July, Orlando, FL, Vol. 3, 2005. pp. 406-411.
30. G.R. Thoma, D.X. Le "Automating data entry for online biomedical databases", *Proc. 14th National Conference on Integrated Online Library Systems IOLS´99*", Medford, NJ, May 1999, pp. 121-128.

31. D.X. Le, L.Q. Tran, et. al., "Automated Medical Citation Records Creation for Web-Based On-Line Journals," *14th IEEE Symposium on Computer-Based Medical Systems*, Bethesda, MD, July 2001, pp. 315-320.
32. D. D. Lewis, "Naïve (Bayes) at Forty: The Independence Assumption in Information Retrieval," Presented at *ECML*, 1998.
33. A. McCallum and K. Nigam, "A Comparison of Event Models for Naïve Bayes Text Classification," AAAI-98 Workshop on Learning for Text Categorization, 1998, pp.577.
34. S. Sohn, W.K. Kim, et. al., "Optimal Training Sets for Bayesian Prediction of MeSH Assignment," Journal of the American Medical Informatics Association, 2008, (Accepted).
35. D.M. Bikel, R.L. Schwartz and R.M. Weischedel, "An Algorithm that Learns What's in a Name," Machine Learning, vol. 34, no. 1-3, pp. 211-231, 1999.
36. E.F. Tjong, K. Sang and F. De Meulder, "Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition," Proc. 7th Conf. Natural Language Learning (CoNLL-2003), pp. 142-147, 2003.
37. J.D. Kim, T. Ohta, Y. Tateishi and J. Tsujii, "Introduction to the Bio-Entity Recognition Task at JNLPBA," Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA), 2004.
38. C. Lee, W. J. Hou and H.-H. Chen, "Annotating Multiple Types of Biomedical Entities: A Single Word Classification Approach," Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA), 2004.
39. B. Settles, "Biomedical Named Entity Recognition Using Conditional Random Fields and Novel Feature Sets," Proc. Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA), 2004.
40. L. Si, T. Kanungo, X. Huang, "Boosting Performance of Bio-Entity Recognition by Combining Results from Multiple Systems," Proc. Workshop on Data Mining in Bioinformatics (BioKDD), 2005.
41. H. Drucker, V. Vapnik and D. Wu, "Automatic text categorization and its applications to text retrieval," IEEE Trans. Neural. Network. 10, 5, 1048–1054, 1999.
42. R.E. Schapire and Y. Singer, "BoosTexter: A boosting-based system for text categorization," Machine Learning, 39(2/3), 135-168, 2000.
43. V. Vapnik, The nature of statistical learning theory, New York: Springer-Verlag, 1995.
44. C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," Data Mining and Knowledge Discovery, 2(2), 1-43 (1998).
45. B. Zhao, Y. Liu, S. Xia, "Support vector machine and its application in handwritten numeral recognition," Proc. 15th Int'l Conf. Pattern Recognition (ICPR) 2, 2720-2723, Barcelona, Spain, Sept. (2000).
46. B. Heisele, T. Serre, M. Pontil, and T. Poggio, "Component-based face detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 1, 657-662, Hawaii, Dec. (2001).
47. T. Joachims, "Text categorization with support vector machine," Proc. Euro. Con. Machine Learning (ECML), 137-142 (1998).
48. F. Sebastiani, "Machine Learning in Automated Text Categorization", ACM Computing Surveys, 34, 1, 2002, 1-47.
49. L. Galavotti, F. Sebastiani and M. Simi, "Experiments on the Use of Feature Selection and Negative Evidence in Automated Text Categorization," Proc. ECDL, pp. 59-68, 2000.
50. C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
51. S. Dumais, J. Platt, D. Heckerman and M. Sahami, "Inductive Learning Algorithms and Representations for Text Categorization," Proc. 7th Conf. Information Retrieval and Knowledge Management, pp. 148-155, 1998.
52. T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," Proc. 10th European Conf. on Machine Learning, pp. 137-142, 1998.
53. E. Leopold and J Kindermann, "Text Categorization with Support Vector Machines. How to Represent Texts in Input Space?" Machine Learning, vol. 46, pp. 423-444, 2002.
54. K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi, "Toward information extraction: identifying protein names from biological papers," Proc. of the Pacific Symposium on Biocomputing'98, 705-716, Jan. (1998).
55. S. Mukherjea, L. V. Subramaniam, G. Chanda, R. Kothari, V. Batra, D. Bhardwaj, and B. Srivastava, "Enhancing a biomedical information extraction system with dictionary mining and context disambiguation," IBM Journal of Research and Development, 48(56), 693-701 (2004).
56. C. Nobata, N. Collier, and J. Tsujii, "Automatic term identification and classification in biology texts," Proc. 5th Natural Language Processing Pacific Rim Symposium, 369-374 (1999).
57. M. A. Andrade and A. Valencia, "Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families," Bioinformatics, 14(7), 600-607 (1998).
58. N. Collier, C. Nobata, and J. Tsujii, "Extracting the names of Genes and Gene products with a hidden Markov model," Proc. The 18th Int'l Conf. Computational Linguistics, 201-207, Saarbrucken, Germany (2000).

59. J. Kazama, T. Makino, Y. Ohta, J. Tsujii, “Tuning support vector machine for biomedical named entity recognition,” Proc. Workshop on NLP in the biomedical domain, 1-8 (2002).
60. C. Sanderson and K. K. Paliwal, “Noise compensation in a person verification system using face and multiple features,” Pattern Recognition, 36(2), 293-302, (2003).
61. S. Rüping, mySVM-Manual, University of Dortmund, Lehrstuhl Informatik 8. 2000 [http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/].
62. C. Nobata, S. Sekine, M. Murata, K. Uchimoto, M. Utiyama, and H. Isahara, “Sentence extraction system assembling multiple evidence,” Proc. 2nd NTCIR Workshop, 319-324 (2001).
63. T. Kikuchi, S. Furui, and C. Hori, “Automatic speech summarization based on sentence extraction and compaction,” Proc. Int’l Conf. Acoustics, Speech, and Signal Processing (ICASSP), I, 384-387, Hong Kong, April (2003).

12. Questions for the Board

1. For the automated extraction of bibliographic data from online journals, what machine learning techniques (other than those we have implemented) may offer higher performance?

2. Are you aware of similar approaches elsewhere for the extraction of bibliographic data from online journals?

3. Do you see any opportunities to extend our current methods to directly assist the indexers?

Glossary

AHCPR	Agency for Health Care Policy and Research
API	Application Programming Interfaces
BAG	Block Adjacency Graph
CDC	Centers for Disease Control and Prevention
CON	Data related to articles commented on (a MEDLINE field)
CSD	Carbohydrate Structure Database
DAN	Databank Accession Number (a MEDLINE field)
DCMS	Data Creation and Maintenance System
DDBJ	DNA Data Bank of Japan
DOM	Document Object Model
EMBL	European Molecular Biology Laboratory
FDA	Food and Drug Administration
FTP	File Transfer Protocol
GDB	Genome Database
GDS	Gene Expression Omnibus Data Set
GEO	Gene Expression Omnibus
GN	Grant Number (a MEDLINE field)
GPL	Gene Expression Omnibus PLatform
GS	Grant Support (a MEDLINE field)
GSE	Gene Expression Omnibus SEries
GSM	Gene Expression Omnibus SaMple
HGML	The Human Gene Mapping Library
HMM	Hidden Markov Model
HTML	Hyper-Text Markup Language
IE	Internet Explorer
ISRCTN	International Standard Randomised Controlled Trial Number
LibSVM	A Library for Support Vector Machines
MYSVM	A Library for Support Vector Machines
NCBI	National Center for Biotechnology Information
NCT	National Clinical Trials
NER	Named-Entity Recognition
OASH	Office of the Assistant Secretary of Health
OMIM	Online Mendelian Inheritance in Man
PDB	Protein Data Bank
PDF	Portable Document Format
PDR	Publisher Data Review
PIR	Protein Information Resource
PMID	PubMed ID
PREFSEQDB	Protein Research Foundation
PubChem	Database for the biological activities of small molecules
RAM	Random Access Memory
RBF	Radial Basis Function
RefSeq	Reference Sequence
SAMHSA	Substance Abuse and Mental Health Services Administration

SVM	Support Vector Machine
SwissProt	Protein Sequence Database
VIPS	VIsion-based Page Segmentation
XML	Extensible Markup Language